CHAPTER-IV

Studies on Hybrid Parallel Computer

Interconnection Network Topology
4.0 Introduction

Advances in science and technology have put high demands for high performance large scale data handling and analysis. Depending upon the area of application or the problem domain the system may be called as parallel and/or distributed [208]. The overall performance of these systems often depends on the effectiveness of its interconnecting network architecture that is the layout pattern of the interconnection of the various components. Consequently, new types of interconnection networks are appearing at an astonishing rate. The main reason for the introduction of so many different IN’s is that no single network provides the optimal performance under all conditions. Each network has its own advantages and drawbacks in terms of the factors such as cost, latency and bandwidth. Thus, it is essential to select a suitable interconnection architecture for a particular application [49, 61]. This requires an extensive study and modification of the existing parallel interconnection architectures.

Most of the interconnection networks discussed so far can be classified as pure networks, meaning that a single set of connectivity rules governs the entire network. The pure networks are modified in two ways namely Hierarchical (Multilevel) or Hybrid (or Composite) networks. The Hierarchical interconnection networks (HIN) are designed through recursive substitution that is each node of the network is replaced with another network. On the other hand, the Hybrid networks combine the connectivity rules from two or more pure networks in order to achieve some advantages from each structure to derive new network sizes that are unavailable with either of the pure architectures or realize any number of performance/cost benefits [173].

Numerous hierarchical combinations of known interconnection networks have been proposed over the past decade. For example, from the class of cube based networks many hierarchical networks are designed namely Hierarchical hypercube (HHC) [152], Hierarchical crossed cube (HCC) [117, 118], Hierarchical folded hypercube network (HFN) [50, 207],
Hierarchical cubic network (HCN) [69], Hierarchical cube connected cycle [211], Pyramided hypercube [41] etc. The HINs are basically of two types namely Homogeneous and Heterogeneous. If the clusters in the HIN at all levels are the same then it is called as Homogeneous otherwise it is termed Heterogeneous [161].

The Star graph a fascinating alternative of Hypercube is also extensively studied in the past to design different hybrid and/or hierarchical networks namely: Star Pyramid [257], Hierarchical star [230], Hyperstar [1], Star connected cycle [17], Starcube [215]. Much of research efforts have been directed to make these networks highly scalable. As a result, the hybrid networks have become an emerging trend and have started receiving much attention in research as well as in commercial sectors. Hybrid networks use a combination of network topologies in such a way that the resulting system becomes a cost effective parallel processing tool solving scientific, engineering and commercial applications. Advances in computational and communication technologies have made hybrid networks economically feasible leading to development of large scale multicomputer systems. Such systems can provide a platform for wide variety of parallel applications. The Hierarchical interconnection network (HIN) is a type of hybrid network which provides a framework for designing networks with reduced link cost by taking advantage of the locality of communication that exists in parallel applications. The HINs employ multiple levels. The Lower-level networks provide local communication while the higher-level networks facilitate remote communication. The HINs may also provide fault tolerance in the presence of some faulty nodes and/or links. The existing HINs can be broadly classified into two classes: those that use nodes and/or links replication and those that use standby interface nodes. The former class includes Hierarchical Cubic Networks, Hierarchical Completely Connected Networks, and Triple-based Hierarchical Interconnection Networks. The latter HINs class includes Modular Fault-Tolerant Hypercube Networks and Hierarchical Fault-Tolerant Interconnection Network. The important topological properties that are considered at the design stage are the network degree, diameter and cost.

In the current chapter three new hybrid parallel interconnection network topologies are proposed namely the Star crossed cube, Meta crossed cube and the Meta star. In the Metacrossed cube and the Meta star network, the seed networks are replaced with new networks so as to improve their topological properties. The Starcrossed cube is designed by combining the Star graph and the Crossed cube.
The next section presents the required background and preliminaries about the seed networks to facilitate the subsequent presentations. In Sections 4.2 to 4.4 the detail structure of the proposed topologies are presented. The topological properties, routing and performance analysis are presented in the corresponding subsections. The results and discussions are presented in Section 4.5. The chapter concludes with remarks in Section 4.6.

4.1 Background

In this section, the Starcube [215], the Hierarchical star [230] and the Hyperstar [1] that form the background for comparison are discussed. The Star graph, Crossed cube and Metacube which form the basis for development of the proposed networks have been already discussed in Chapters 3 and 2 in Subsections 3.1.5, 2.1.3 and 2.1.5 respectively and therefore do not need further elaboration at this stage.

4.1.1 Notation

\begin{itemize}
  \item \textbf{D(G)} Diameter of graph G
  \item \( \bar{d} \) Average node distance
  \item \textbf{E} Set of edges
  \item \( k \) Dimension of Crossed cube
  \item \( n \) Dimension of Star graph
  \item \( N_d \) Number of nodes at distance \( d \)
  \item \( p \) Total number of processors
  \item \( R_l \) Link reliability
  \item \( R_p \) Node reliability
  \item \( t \) Mission time in Hrs.
  \item \textbf{TR} Terminal reliability
  \item \( \lambda_l \) Link failure rate
  \item \( \lambda_p \) Processor failure rate
  \item \( \xi \) Cost of the network
  \item \( \rho \) Ratio of link cost to processor cost
  \item \( \eta \) Message traffic density
  \item \( \alpha \) Time penalty
  \item \( \sigma \) Ratio of costs between penalty and processor
\end{itemize}
4.1.2 Starcube Network

The Starcube denoted as SC(n,m) is a product network resulted from the Cartesian product of two networks: the Star graph of dimension 'n' and the hypercube of dimension 'm' [215]. This network is developed with an aim to bridge the gap between two consecutive sizes of the n-Star. Being a product graph, the SC(n,m) possesses all the attractive properties of the star and cube topology. The SC(n,m) is vertex as well as edge symmetric graph. The individual nodes in the Starcube are addressed using two parameters <x,y> where x is called the hypercube part address and y is the star part address. Each node in it has two types of neighbors namely star part neighbor and the hypercube part neighbor. The Starcube SC(3,2) is shown in Fig. 4.1.

![Figure 4.1: Starcube of degree 4, SC(2,3)](image)

4.1.3 Hierarchical Star

The Hierarchical star denoted as HS(n,n) or HS_n is a two level interconnection network topology designed using n-star graph as its basic building blocks [230]. Thus HS_n consists of n! modules where n ≥ 3. Each node in the HS is denoted by a two-tuple address (x, y) where x and y are arbitrary permutations of 'n' distinct symbols. For each node (x, y), x identifies the module
address and \( y \) identifies the address of the node within that module. There are two types of edges in \( HS_n \) namely local links and external links. The nodes within the module are connected through the local links whereas the individual building blocks are interconnected using external links.

The \( HS_n \) is a regular graph of degree \( n \) containing \( (n!)^2 \) number of nodes. It's diameter is at most \( (3n - 2) \). Though \( HS \) is a very large scale network, it still suffers from the same drawback the Star graph has faced. The network grows at a very faster rate as compared to the star graph itself. Again as the topology suggested, the \( HS \) is characterized by a single parameter \( n \). When \( n \) is 3 the network size is 36 but when \( n \) is 4, the network contains 576 nodes. This significant gap in the two consecutive sizes of Hierarchical Star becomes a major disadvantage. Another disadvantage of Hierarchical star is that the dimension cannot take any values of \( n \) like Metacube. It only takes values like (3,3), (4,4), (5,5) etc. In Metacube the dimension parameters \( k \) and \( m \) can take any values (1, 2), (1,3), (1,4), (2,3), (2,4) and so on. Thus the Meta cube is a suitable candidate for variable node size applications.

![Figure 4.2: Hierarchical star of dimension 3, HS(3,3)](image)

The Hierarchical star of dimension three is shown in Fig. 4.2. In the Figure the basic modules are shown with their corresponding addresses. Individual nodes addresses are not shown.
4.1.4 Hyperstar

The interconnection networks are always preferred with small degree and diameter as it results in reduced cost. The Hyperstar graph has been proposed as a better alternative to Star graph with reduced cost. The Hyperstar graph is formally defined as follows:

The Hyperstar\( (k,n) \) is a graph \( G(V,E) \) with node set \( V \) and edge set \( E \). Any node \( v \in V \) is represented by a string of \( n \) bits \((v_1 v_2 \ldots v_n )\), \( v_i \in \{0,1\} \), in which \( k \) (\( n > k \)) bits are 1 and the rest \( (n-k) \) bits are 0's. Two nodes \( u = (v_1 v_2 \ldots v_{i-1} v_i v_{i+1} \ldots v_n) \) and \( v = (v_1 v_2 \ldots v_{i-1} v_i v_{i+1} \ldots v_n) \) are connected by an edge \((u,v) \in E\) if and only if \( v_1 \) is complement of \( v_2 \) and the bit string is obtained by exchanging \( v_1 \) and \( v_i \), \( 2 \leq i \leq n \) in the bit string of \( u \). The total number of nodes in Hyperstar\((n,k)\) is \( \binom{n}{k} \). The node degree is \( k \) and diameter is \( (n-1) \) if \( n=2k \). It is a regular network if \( n=2k \) otherwise it is neither symmetric nor regular. In that case the structure of the network is bit complicated as the nodes do not have same node degree. The network of degree 3 is shown below in Fig. 4.3.

![Figure 4.3: The Hyperstar(6; 3) generated by connecting Hyperstar(5; 2) with Hyperstar(5; 3)](image-url)
In the following sections three new parallel interconnection networks namely Star crossed cube, Meta crossed cube and Meta star are proposed.

4.2 Proposed Topology: Star Crossed Cube (SCC)

The proposed topology is based on the Crossed cube [51] and the Star graph $S_n$ or $n$-star [12]. The network structures of both the crossed cube and star graph are briefly discussed previously in Chapter II and Chapter III respectively of this Thesis.

4.2.1 Construction of SCC

The Star crossed cube denoted by SCC $(m,n)$ is the product graph of an $m$-dimensional $CC(m)$ and an $n$-dimensional star $S(n)$. That is in an $n$-Star ($S_n$), each vertex is replaced with a $m$-dimensional Crossed cube. Then, the node address of each vertex in the resulting graph will have two parts $<x_m,1,x_m,2,...,x_0,y_0,y_1,...,y_n,1>$ where $x_i$’s represent the crossed cube part and $y_i$’s represent the star part. Each node will have two types of neighbors namely crossed cube part neighbor and star part neighbor with node address $<x_{m-1},x_{m-2},...,x_0,y_0,y_1,...,y_n,1>$, $<x_{m-1},x_{m-2},...,x_0,y_0,y_1,...,y_n,1>$ respectively.

The SCC(3,3) is shown below in Fig. 4.4. The SCC (3,3) has six sub modules. Two individual sub modules of SCC(3,3) are shown in Fig. 4.4 (a) and (b) respectively. The first module has ‘123’ as its Star part label. Similarly the second module has ‘213’ as its Star part label. The complete network with all the modules is shown in Fig. 4.4 (c). The Fig. 4.4 (a) shows the first sub module where all the eight nodes have star part label as 123 and corresponding labels of Crossed cube. Similarly Fig. 4.4 (b) shows the second sub module of SCC(3,3) where the nodes have 213 as their star part level. The Figure 4.6 (a) and (b) show the Star crossed cube of dimension 6 and 7 respectively. In Fig. 4.6 (b) some of the links from the 5th cluster to rest of the clusters are shown. The nodes in this cluster will have a five digit permutation such as 12345 as their star part label.
Figure 4.4: Star crossed cube and its basic modules, a) SCC_{123} and b) SCC_{213}, c) SCC(3,3)
4.2.2 Topological Properties of SCC

The following are the various topological properties of the proposed Star crossed cube network.

**Theorem 4.1:** The total number of nodes in SCC(m,n) graph is given by

\[ p = n! \times 2^m \]  

(4.1)

**Proof:** In an n-Star the total number of nodes is \( n \). In a Crossed cube the total number of nodes is \( 2^m \). SCC being a product graph, the total number of nodes is equal to \( p = n! \times 2^m \). Hence the result.

**Theorem 4.2:** The total number of edges in SCC(m,n) is given by

\[ E = n! \times 2^{m-1} (m + n - 1). \]  

(4.2)

**Proof:** In SCC(m,n) there are \( n! \) number of CCm’s connected in n-Star topology. So \( n! \) numbers of Crossed cubes will have \( n! \times m \times 2^{m-1} \) number of edges. Next, each Crossed cube is connected to \( (n-1) \) neighbors. So, the total number of edges interconnecting \( n! \) Crossed cubes is \( n! \times \left( \frac{n-1}{2} \right) \times 2^m \).

Hence, the total number of edges in SCC(m,n) is

\[ E = n! \times m \times 2^{m-1} + n! \times \left( \frac{n-1}{2} \right) \times 2^m \]

\[ = n! \times 2^{m-1} (m + n - 1). \]

**Theorem 4.3:** The degree of SCC(m,n) is \( (m+n-1) \).

(4.3)

**Proof:** In a Crossed cube the degree of each node is \( m \). Next there are \( (n-1) \) edges incident on each vertex to form the n-Star. Hence, the degree of each vertex in SCC(m,n) is \( (m+n-1) \).

**Theorem 4.4:** The diameter of SCC(m,n) is \( D(G) = \left\lceil \frac{3(n-1)}{2} \rightceil + \left\lceil \frac{m+1}{2} \right\rceil \).

(4.4)

**Proof:** Let \((u, v)\) and \((u', v')\) be two nodes in SCC \((m, n)\). Then, traveling from \((u, v)\) to \((u', v)\) takes at most \( \left\lceil \frac{m+1}{2} \right\rceil \) steps as these two nodes are in the same Crossed cube of order \( m \). Next, traveling from \((u', v)\) to \((u', v')\) takes at most \( \left\lceil \frac{3(n-1)}{2} \right\rceil \) steps that is the diameter of the Star graph; as the nodes with the same Crossed cube part label form an n-Star. Hence, in atmost \( \left\lceil \frac{3(n-1)}{2} \right\rceil + \left\lceil \frac{m+1}{2} \right\rceil \) steps the node \((u', v')\) can be reached from \((u, v)\) that is the diameter of SCC\((m,n)\).
Chapter IV  

Some Studies on Parallel Computer Interconnection Network Topology

**Theorem 4.5:** The cost of the SCC network is $\xi = (m + n - 1) \left( \frac{3(n-1)}{2} + \frac{m+1}{2} \right)$.  

*Proof:* In symmetric networks the cost factor $\xi$ is defined as the product of the node degree and diameter. From Theorem 3, the degree of SCC is $(m + n - 1)$. Again from Theorem 4.4 the diameter is $\left( \frac{3(n-1)}{2} + \frac{m+1}{2} \right)$. Hence the cost is given by  

$$\xi = (m + n - 1) \left( \frac{3(n-1)}{2} + \frac{m+1}{2} \right)$$  

**Theorem 4.6:** The average distance of SCC(m,n) is  

$$d_{sec} = \frac{11x + 4y}{8} + n - 4 + \frac{2 + \sum_{i=1}^{n-1} \frac{1}{i}}{n}$$  

(4.6)  

where $m = 3x + y$, $y < 3$ and $x, y$ are integer values.

*Proof:* The average distance of Crossed cube is $\frac{11x + 4y}{8}$ and for Star graph it is $n - 4 + \frac{2 + \sum_{i=1}^{n-1} \frac{1}{i}}{n}$. Hence the result follows.

**Theorem 4.7:** The message density of Star crossed cube is given by  

$$\eta = \frac{2d_{sec}}{(m+n-1)}$$  

(4.7)  

*Proof:* This message density is defined as $\eta = \frac{dp}{E}$, where $p$ is the total number of nodes, $d$ is the average node distance and $E$ is the total number of links. It is assumed that each node is sending one message to a node at distance $d$ on the average.

From Theorem 4.6,  

$$d_{sec} = \frac{11x + 4y}{8} + n - 4 + \frac{2 + \sum_{i=1}^{n-1} \frac{1}{i}}{n}$$

From Proof of Theorem 4.1 and 4.2, we get  

$$p = n! \times 2^m \quad \text{and} \quad E = n! \times 2^{m-1}(m + n - 1)$$

Hence, the message density is given by  

$$\eta = \frac{n! 2^m \left(\frac{11x + 4y}{8} + n - 4 + \frac{2 + \sum_{i=1}^{n-1} \frac{1}{i}}{n}\right)}{n!2^{m-1}(m+n-1)}$$

$$= \frac{(\frac{11x + 4y}{8} + n - 4 + \frac{2 + \sum_{i=1}^{n-1} \frac{1}{i}}{n})}{(\frac{m+n-1}{2})}$$

Hence, the message traffic density is given by $\eta = \frac{2d_{sec}}{(m+n-1)}$
Chapter IV

Some Studies on Parallel Computer Interconnection Network Topology

Theorem 4.8: Partitioning of SCC(m,n) can be done into two SCC(m-1, n)’s in m ways and into two SCC(m, n-1)’s in(n-1) ways.

Proof: From the system architecture of SCC in Fig. 4.4 it is evident that the SCC consists of crossed cubes which are connected in star graph fashion. The CC(m)’s can be partitioned into CC(m-1)’s in m ways. The skeleton of SCC is a n-star which can be partitioned into (n-1)-stars in (n-1) ways.

Hence the result.

Theorem 4.9: The SCC(m,n) has (m + n - 1) number of node disjoint paths between any two of its nodes.

Proof: In a Crossed cube of dimension m, the node connectivity is m. Similarly in an n-Star it is (n - 1). So in SCC the node connectivity becomes (m + n - 1). Hence, there are (m + n - 1) number of node disjoint paths between any two nodes.

Different topological properties of the proposed topology are compared with that of the parent networks in Table 4.1.

Table 4.1: Comparison of topological properties of SCC

<table>
<thead>
<tr>
<th>Parameters</th>
<th>HC</th>
<th>CC</th>
<th>Star</th>
<th>Starcube</th>
<th>SCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nodes</td>
<td>$2^n$</td>
<td>$2^n$</td>
<td>$n!$</td>
<td>$n!2^m$</td>
<td>$n!2^m$</td>
</tr>
<tr>
<td>Links</td>
<td>$m2^{m-1}$</td>
<td>$m2^{m-1}$</td>
<td>$n!\left(\frac{n-1}{2}\right)$</td>
<td>$n!2^{m-1}(m+n-1)$</td>
<td>$n!2^{m-1}(m+n-1)$</td>
</tr>
<tr>
<td>Degree</td>
<td>$m$</td>
<td>$m$</td>
<td>$n-1$</td>
<td>$m+n-1$</td>
<td>$m+n-1$</td>
</tr>
<tr>
<td>Diameter</td>
<td>$\frac{m+1}{2}$</td>
<td>$\frac{3}{2}(n-1)$</td>
<td>$\frac{m+1}{2}\left[\frac{3}{2}(n-1)\right]$</td>
<td>$\frac{m+1}{2}\left[\frac{3}{2}(n-1)\right]$</td>
<td></td>
</tr>
<tr>
<td>Avg. Dist.</td>
<td>$\frac{m}{2}$</td>
<td>$\frac{11x+4y}{8}$</td>
<td>$\frac{2}{n} + \sum_{i=1}^{n} 1/i$</td>
<td>$\frac{m+n-4 + \frac{2}{n}}{n} + \sum_{i=1}^{n} 1/i$</td>
<td>$\frac{11x+4y}{8} + n-4 + \frac{2}{n} + \sum_{i=1}^{n} 1/i$</td>
</tr>
<tr>
<td>Cost</td>
<td>$m^2$</td>
<td>$m\left[\frac{m+1}{2}\right]$</td>
<td>$(n-1)\frac{3}{2}(n-1)$</td>
<td>$(m+n-1)(m + \frac{3}{2}(n-1))$</td>
<td>$(m+n-1)\left[\frac{m+1}{2} + \frac{3}{2}(n-1)\right]$</td>
</tr>
<tr>
<td>Msg. Traffic Density ($\eta$)</td>
<td>$\frac{2d_{sc}}{n}$</td>
<td>$\frac{2d_s}{n-1}$</td>
<td>$\frac{2d_{sc}}{(m+n-1)}$</td>
<td>$\frac{2d_{sc}}{(m+n-1)}$</td>
<td></td>
</tr>
</tbody>
</table>
4.2.3 Routing and Broadcasting in SCC

The problem of finding a path from a source node $s$ to a destination node $t$, and forwarding the message along the path is referred to as Routing. For any parallel processing network it is the most important issue. The routing should be fast, simple and cost effective. The current section describes the routing algorithm for the proposed SCC network.

The Crossed cube and Star graph both have self-routing algorithms. So, the SCC which is a hybrid network of Crossed cube and Star also allows self-routing. The routing algorithm for the proposed SCC network is described as follows:

Let the source and destination nodes are $u(c, s)$ and $v(c', s')$ respectively, where $c$ and $s$ represent the cube part label and the star part label of the source node $u$. Similarly $c'$ and $s'$ represent the cube part and star part label of the destination node $v$ respectively.

**Procedure SCC Routing** $(m, n, u, v)$

**begin**

**Step 1:** Route the message from $u$ to $u_1$. Address of $u_1$ will be either $(c', s)$ or $(c, s')$. For the first case Crossed cube routing will be used as discussed in [27]. In the second case Star routing is used.

**Step 2:** Then from $(c', s)$ the destination node $(c', s')$ can be easily reached using star routing. If the intermediate node is $(c, s')$ then the crossed cube self routing will be used to reach the destination node that is $(c', s')$.

**end**

Next an optimal algorithm for one-to-all broadcasting algorithm is proposed for the proposed Star crossed cube network. For this algorithms the Star-broadcast algorithm [154] is used as a subroutine. Suppose $u(c, s)$ be an arbitrary node in any cluster of SCC. The following algorithm broadcasts a message to all other nodes in the SCC$(m, n)$.

**Procedure SCC One-to-all Broadcast**

**begin**

**Step 1:** suppose the cluster $(c, *)$ contains the source node $u(c, s)$. Then using the CC-broadcast algorithm message can be transmitted to all other nodes within the cluster.

**Para do**

**begin**

116
Step 2: next for inter cluster communication the star links are used in which each node (x,v) of cluster (c,*c) transmits the message to node (x,v) using star-broadcast. Then in the other clusters at least one node will be loaded with the message.

Step 3: now the loaded nodes in each cluster will do a broadcast within the cluster using CC-broadcast algorithm.

Theorem 4.10: The one-to-all broadcast algorithm for SCC(m,n) takes $O(m + n \log n)$ time.

Proof: The three steps in the above algorithm take $O(m)$, $O(n \log n)$ and $O(m)$ time respectively. Thus, the entire algorithm takes $O(m + n \log n)$ time.

Algorithm: All-to-all Broadcast

In all-to-all broadcasting every node in the network sends a message to all other nodes in the network. So the one-to-all algorithm can be executed for each node for complete broadcast of the message in the network.

Theorem 4.11: In SCC(m,n) the time complexity for all-to-all broadcast is $O(M + n \log n)$.

Proof: For a cube type network of dimension $m$, consisting of $M$ nodes, the time complexity for all-to-all broadcasting is $O(M)$ [71,154]. In an n-Star, any sequence of nodes that constitutes a one-to-all broadcast algorithm is also all-to-all broadcasting algorithm.

Hence, in SCC(m,n), the time complexity of the all-to-all broadcasting algorithm is $O(M + n \log n)$.

4.2.4 Embedding of Other Architectures in SCC

The embedding of other architectures is regarded as an important property of any interconnection topology. Embedding of rings, meshes and binary trees into Star graphs [18, 85,106, 123, 187, 217, 246] and crossed cubes are previously studied in [47, 56, 57]. This subsection investigates the embedding of rings and meshes into the proposed Star crossed cube. The following paragraphs explain the embedding of various topologies in SCC.

The embedding of a guest graph $G(V,E)$ into another host graph $S$ is a mapping of the set of vertices $V(G)$ into that of $S$, that is $V(S)$ and of $E(G)$ into $E(S)$. The mapping is denoted by $R$. Thus any vertex $x$ of $G$ is mapped through $R(x)$ of $S$ uniquely. That means $R(x) \neq R(y)$ for $x \neq y$.
and \( x, y \in V(G) \). However, for the embedding to exist \(|S| \geq |G|\) and the host graph \( S \) has to be a connected one. The ratio \(|S|/|G|\) is called the expansion of the embedding [72].

The dilation is defined as follows: \( \text{Dilation} (d) = \max \{ \text{length of the shortest path from } R(x) \text{ to } R(y) \} \). The congestion of any edge \( e \) of \( S \) is the number of paths (each path representing an edge of \( G \) mapped to \( S \)) of \( G \) which contains \( e \). The maximum of the congestions of all edges of \( S \) is the congestion of the embedding.

### 4.2.4.1 Embedding of Ring in SCC

A ring can be embedded into SCC\((m,n)\) using Grey Codes. A Grey code is a well known sequence of binary bits where two consecutive codes differ by only one bit. The one bit Grey code is \( G_1 = (0, 1) \). From \( G_1 \), the two bit Grey code \( G_2 \) can be derived as \((00, 01, 11, 10)\). For \( n > 2 \), \( G_n = (0 G_{n-1}, 1 G_{n-1}) \), where \( G_n' \) represents the reverse string of \( G_n \).

So \( G_3 = (0 G_2, 1 G_2') = (000, 001, 011, 010, 110, 111, 101, 100) \). Next \( G_4 = (0 G_3, 1 G_3') \) and \( G_5 = (00 G_3, 01 G_3', 11 G_3', 10 G_3) \). The main significance of this code is that the first and last labels differ by one bit only.

A ring can be successfully embedded in SCC. As shown in Fig. 4.5, the arrow heads show the sequence of embedding. A ring with eight nodes bearing the codes as in \( G_3 = (0, 1, 3, 2, 6, 7, 5, 4) \) can be easily embedded in SCC\(_{123} \), the first sub module as defined above in Fig. 4.3 (a).

\[ R(0) = (0, 123), R(1) = (1, 123), R(3) = (3, 123) \ldots \text{, and } R(4) = (4, 123). \]

For a sixteen node ring with codes as in \( G_4 = (0, 1, 3, 2, 6, 7, 5, 4, 12, 13, 15, 14, 10, 11, 9, 8) \) the embedding extended to two sub modules SCC\(_{123} \) and SCC\(_{213} \) is as follows:

\[ R(0) = (0, 123), R(1) = (1, 123), R(3) = (3, 123) \ldots \text{, and } R(4) = (4, 123) \]
\[ R(12) = (4, 213), R(13) = (5, 213), R(15) = (7, 213), R(14) = (6, 213) \ldots \text{, and } R(8) = (0, 213). \]

Obviously, the dilation and expansion for this embedding are all 1.
Chapter IV Some Studies on Parallel Computer Interconnection Network Topology

Figure 4.5: Ring embedding in sub modules SCC_{123} and SCC_{213}

4.2.4 Embedding of Mesh in SCC

Generally in a Crossed cube of dimension $m$, a family of disjoint three dimensional meshes of size $2 \times 2 \times 2^{m-3}$ can be embedded for $m \geq 4$ [47]. The current work illustrates embedding of three dimensional meshes into the SCC topology.

**Theorem 4.12:** For $m \geq 3$, a family of $2^{m-2}(n)(n-1)(n-2) \ldots 3$ disjoint three dimensional meshes of size $2 \times 2 \times 2^{m-3}$ can be embedded into SCC($m,n$) with unit dilation and unit expansion. The Theorem is proved by induction method. Before that some Lemmas are proposed in support of the Theorem 4.12.
Lemma 4.1: In SCC(3,3), a family of $2^{3-2} \times 3 = 6$ disjoint three dimensional meshes can be embedded.

Proof: In SCC(3,3), the sub graph SCC(CC3,123) will combine with SCC(CC3,321). Then SCC(CC3,213) will combine with SCC(CC3,312) and SCC(CC3,231) with SCC(CC3,132). Thus there are 3 sets of interlinks and each interlink forms 2 disjoint meshes. So the number of meshes embedded = 2 x 3= 6. The meshes are as follows:

$\begin{align*}
&m_{11} = \begin{pmatrix} 0,123 & 2,123 \\ 0,321 & 2,321 \end{pmatrix}, \quad m_{12} = \begin{pmatrix} 4,123 & 6,123 \\ 4,321 & 6,321 \end{pmatrix}, \\
&m_{21} = \begin{pmatrix} 0,231 & 2,231 \\ 0,132 & 2,132 \end{pmatrix}, \quad m_{22} = \begin{pmatrix} 4,231 & 6,231 \\ 4,132 & 6,132 \end{pmatrix}, \\
&m_{31} = \begin{pmatrix} 0,213 & 2,213 \\ 0,312 & 2,312 \end{pmatrix}, \quad m_{32} = \begin{pmatrix} 4,213 & 6,213 \\ 4,312 & 6,312 \end{pmatrix}
\end{align*}$

The cube part node addresses are represented in decimal numbers that is 0, 1, 2, … 7.

This embedding has unit dilation and unit expansion.
**Lemma 4.2:** In an SCC(4,3), a family of \((2^{4-2} \times 3) = 12\) disjoint three dimensional meshes can be embedded.

*Proof:* In SCC(4,3), the CC_4 is the basic module in the Star of dimension 3 (3-Star). So there are six CC_4's in total. Each Crossed cube can embed disjoint 3D meshes [47]. Again 3 sets of interlinks can be used in the 3-Star. Each interlink can form 4 disjoint meshes. Hence, the number of meshes embedded in a SCC(4,3) = \((2^{4-2} \times 3) = 12\). The meshes are as follows:

\[ M_1 = (0(m_{11}) 0(m_{12}) 1(m_{12}) 1(m_{11})) \]  
\[ M_2 = (0(m_{21}) 0(m_{22}) 1(m_{22}) 1(m_{21})) \]  
\[ M_3 = (0(m_{31}) 0(m_{32}) 1(m_{32}) 1(m_{31})) \]

**Lemma 4.3:** In SCC(3,4), a family of \((2^{3-2} \times 4 \times 3) = 24\) disjoint three dimensional meshes can be embedded.

*Proof:* Here, the basic building blocks are CC_3 on a 4-Star as shown in Fig. 4.6 (a). So there can be \(4! = 24\) crossed cubes in total. Then the number of interlinks is \(n(n-1) = 4\times3\). In SCC(3,4) each set of interlink forms \(2^{3-2}\) number of meshes. In total there can be \((2^{3-2} \times 4 \times 3) = 24\) number of disjoint meshes.

**Lemma 4.4:** In SCC(4,4) a family of \((2^{4-2} \times 4 \times 3) = 48\) disjoint three dimensional meshes can be embedded.

*Proof:* In SCC(4,4) the CC_4 becomes the building block on a 4-Star. So there are 24 crossed cubes. These can be connected using \(n(n-1) = 4 \times 3 = 12\) number of interlinks. Now each set of interlink can form \(2^{4-2} = 4\) meshes. Hence in total, there can be 48 number of meshes embedded.

**Lemma 4.5:** In SCC(3,5), a family of \((2^{3-2} \times 5 \times 4 \times 3) = 120\) number of disjoint three dimensional meshes can be embedded.

*Proof:* In SCC(3,5), the basic building blocks are three dimensional Crossed cubes on a 5-Star. So, there can be \(5! = 120\) crossed cubes. In SCC(3,5), there are five clusters as shown in Fig. 4.6 (b). So the number of interlinks is \(5 \times 4 \times 3 = 60\). Now each set of interlink can form a total of \(2^{3-2} = 2\) meshes. Hence in total there can be \(60 \times 2 = 120\) number of disjoint meshes embedded.
Figure 4.6 (b): Star crossed cube of dimension 7, SCC(3,5)
Proof of Theorem 4.12: In SCC(m,n), the m-dimensional Crossed cube becomes the basic building block on an n-Star graph. By induction on m and n, according to lemma 4.1 to 4.5, CC_m can admit $2^{m-2}$ number of meshes. Then, in the r-Star there are n clusters and each cluster is an (n-1)-Star. So the number of interlinks is $n(n-1)(n-2)...3$. Thus the total number of disjoint 3D meshes in SCC(m,n) is $2^{m-2} \times n(n-1)(n-2)...3$.

Hence the result.

4.2.5 Performance Analysis of SCC

In this section the important measures of performance of the proposed network namely: The Cost Effectiveness Factor (CEF), the Time Cost Effectiveness Factor (TCEF) and Reliability are analyzed. The Cost Effectiveness factor is defined as the ratio of cost effectiveness and efficiency. It takes into account the cost of the system (that is processor cost and link cost) as well as processor utilization [197].

Theorem 4.13: The cost effectiveness factor of SCC(m,n) is

$$CEF(p) = \frac{1}{1 + p \frac{m+n-1}{2}}$$

where $p$ is the ratio of link cost to the processor cost.

Proof: In general the number of links is a function of the number of nodes in the system. Cost Effectiveness is a product of two terms, one refers to the architectural features and the other corresponds to efficiency of the algorithm. In the proposed network, the number of nodes is given by

$$p = 2^m n!.$$

The total number of links is given by

$$E = 2^{m-1} n! (m + n - 1)$$

$$= n! 2^m \left(\frac{m+n-1}{2}\right) = p \frac{m+n-1}{2}$$

$$= f(p).$$

$$g(p) = \frac{f(p)}{p} = \frac{m+n-1}{2}$$

Hence,

$$CEF(p) = \frac{1}{1 + p g(p)}$$

$$= \frac{1}{1 + p \frac{m+n-1}{2}}$$

Theorem 4.14: The time-cost effectiveness factor of the Starcrossed cube network is given by

$$TCEF = \frac{1}{1 + p \frac{m+n-1}{2} + \frac{\sigma}{2^m n!}}.$$  

Proof: The TCEF is given by

123
Chapter IV

Some Studies on Parallel Computer Interconnection Network Topology

\[ TCEF(p, T_p) = \frac{1 + \alpha T_1^{\alpha - 1}}{1 + \rho g(p) + \frac{T_1^{\alpha - 1} \sigma}{p}} \]

where \( T_1 \) 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 multiprocessor system having \( p \) processors and \( \rho \) is the cost of links / cost of processors [197].

As it is seen in Theorem 4.13 above, \( g(p) = \frac{m+n-1}{2} \) and \( \alpha \) is assumed to be \( 1 \) for linear time penalty.

Hence,

\[ TCEF = \frac{1 + \sigma}{1 + \rho g(p) + \left(\frac{\sigma}{p}\right)} = \frac{1 + \sigma}{1 + \rho \left(\frac{m+n-1}{2}\right) + \left(\frac{\sigma}{p}\right)} \]

Reliability Evaluation in SCC:

In SCC\((m,n)\), there are \((m+n-1)\) number of node disjoint paths. First all the possible node disjoint paths in different SCC\((m,n)\)'s are derived and then the two terminal reliability between the two farthest nodes is evaluated. In SCC the lowest possible node degree is four with \( m=2 \) and \( n=3 \). In degree four SCC the two farthest nodes are \(<00, 123 > \) and \(<11,132 > \) and the possible node disjoint paths between them are as follows:

\(<00,123>-<00,213>-<00,312>-<00,132>-<01,132>-<11,132>
<00,123>-<01,123>-<11,123>-<11,213>-<11,132>-<11,132>
<00,123>-<10,123>-<10,321>-<10,231>-<10,132>-<11,132>
<00,123>-<00,321>-<00,231>-<01,231>-<11,23>-<11,132>

Here number of links is five and number of nodes is four.

Hence according to Theorem 2.11, the two terminal reliability for SCC\((2,3)\) is given by

\[ TR(SCC) = 1 - (1 - R_0^5 R_f^4)^4 = 0.9828 \]

In degree five SCC the two farthest nodes are \(<000,123 > \) and \(<111,132 > \) and there are five node disjoint paths possible. They are as follows:

\(<000,123>-<001,123>-<001,213>-<011,213>-<111,312>-<011,132>-<001,132>-<111,132>
<000,123>-<010,123>-<010,213>-<010,312>-<110,312>-<110,132>-<111,132>
<000,123>-<000,321>-<000,231>-<001,231>-<111,231>-<111,132>
<000,123>-<000,213>-<000,312>-<001,312>-<111,312>-<111,132>

Hence two terminal reliability for SCC\((3,3)\) is given by

124
Chapter IV

Some Studies on Parallel Computer Interconnection Network Topology

$$TR\,(SCC) = 1 - \left( \left( 1 - R_1^5 R_2^4 \right) \left( 1 - R_1^6 R_2^5 \right) \right) \left( 1 - R_1^7 R_2^6 \right)$$

$$= 0.9882$$

4.3 Proposed Topology: Meta Crossed Cube (MCC)

In this section attempt has been made to elaborate the detail construction of the second proposed network, the Meta crossed cube (MCC).

4.3.1 Construction of MCC

The proposed Meta crossed cube network, MCC(k,m), has a two layered structure. There are $2^k$ classes. Each class contains $2^m(2^k-1)$ clusters and each cluster contains $2^m$ nodes. These nodes form a Crossed cube of order $m$, $CC_m$ at the lower level. In the higher level the connections are made using the pair related relation [51,53] of the Crossed cube. Thus the higher level structure is again a Crossed cube.

Addressing:

In the MCC network, each node is identified by $n = (m2^k + k)$ bit node address. The $(m2^k + k)$ bit address will have three parts namely (i) $k$-bit class address, (ii) $m(2^k - 1)$ bit cluster address, and (iii) $m$ bit node address. Let $h = 2^k$. The node address has the format: $<c, m_{h-1}, ..., m_1, m_0>$, where $c$ symbolizes the $k$-bit class address, $m_j$ stands for $m$ bit node address and $(m_{h-1}, ..., m_{j+1}, m_{j-1}, ..., m_0)$ stands for $m(h-j)$ bit cluster address.

Within a cluster, the $m$-bit node address forms an $m$ order crossed cube with $m$ links called cross cube edges. These nodes are connected if their node addresses are pair related or if there exists difference in one bit position. There is no link among the nodes when they belong to the same class. For any two nodes whose addresses differ only in one bit position in the class field there is a direct link between them. Some of these links also follow pair related relationship [51].

Thus, in MCC network some of the connections are changed as compared to MC(k,m) due to their pair related relation of crossed cube network topology. The MCC networks of dimension 3 and 4 are shown in Fig. 4.7 and 4.8 respectively. The details of MCC(1,2) is illustrated below.

Illustration

Let $k=1$ and $m=2$ then MCC(1,2) is shown in Fig.4.7. The node addresses are given in equivalent decimal notation in the figure for better clarity. The node addresses have $a (mh+k) = 5$
bit representation in binary. In the equivalent Metacube that is Dual cube(1,2) the neighbours are 
(1,17),(3,19),(5,21),(7,23),(9,25), (11,27),(13,29),(15,31).

Due to the change in links in Metacrossed cube the new neighbours in MCC(1,2) are listed
as follows: (1,19),(3,17),(5,23),(7,21),(9,27),(11,25), (13,31),(15,29) and are shown below in
Fig.4.7.

This change results in reduction of the diameter. This improvement in diameter is better
realised when the value of \( m \) is three or more.

![Metacrossed cube of degree 3, MCC(1,2)](image)

Figure 4.7: Metacrossed cube of degree 3, MCC(1,2)

In the next subsection the topological details of the proposed Metacrossed cube network
are presented.

4.3.2 Topological Properties of MCC

Generally the topological properties such as degree, diameter, cost, bisection width etc.
determine the importance of a network topology. In this subsection different parameters of Meta
crossed cube are evaluated. They are also compared against the parameters of the parent networks
in Table 4.2.
Chapter IV Some Studies on Parallel Computer Interconnection Network Topology

Degree:

The degree of a node in a graph is defined as the total number of edges connected to that node. Also, the degree of a network is defined as the largest degree of all the vertices in its graph representation. The proposed topology MCC is symmetric and the number of edges per node is \((k+m)\). Hence, the degree of MCC network is \((k+m)\).

Nodes:

Theorem 4.15: The total number of nodes of the MCC network is given by

\[
p = 2^{mh+k}, \tag{4.10}
\]

where \(h=2^k\).

Proof: As per the construction, each node in the MCC network is represented by \((mh+k)\) number of binary bits. Hence, the total number of nodes in MCC network is given by

\[
p = 2^{mh+k}
\]

Figure 4.8: Metacrossed cube of degree four, MCC(1,3)
Chapter IV  
Some Studies on Parallel Computer Interconnection Network Topology

Edges:

Theorem 4.16: Total number of edges in Meta crossed cube is \(2^{m+h+k}\left(\frac{m+k}{2}\right)\) \(\quad(4.11)\)

Proof: In Meta crossed cube, the number of nodes is given by \(2^{m+h+k}\), where \(h=2^k\). The number of links per node is \((m+k)\). As the total number of edges in cube type networks is given by
\[E=(\text{total number of nodes})(\text{links per node}/2)\]
Hence,
\[E=2^{m+h+k}\left(\frac{m+k}{2}\right)\]

Diameter:

The diameter of a network is defined as the maximum of shortest distance among all pairs of nodes.

Theorem 4.17: The diameter of the Metacrossed cube is \(D(MCC) = \left[\frac{m+1}{2}\right] + 1 \right) 2^k\) \(\quad(4.12)\)

Proof: The diameter of an MC(k,m) is \((m+1)2^k\), where \(m\) is the diameter of the \(m\)-cube. The diameter of the Meta crossed cube is derived considering the following three cases.

Let \(s\) and \(t\) be two nodes in MCC(k,m). The node address of \(s\) and \(t\) differ in \(r\) bit positions. Now \(s\) and \(t\) both may be in (i) same cluster and same class or (ii) different class or (iii) same class but in different clusters. For case (i) and (ii) the diameter will be similar to that of the crossed cube. But for case (iii), \(2^k\) number of edges will be added as it has to pass through other classes in order to reach the cluster of the same class.

Since there are \(2^k\) classes, so traversing within each low level crossed cube will require a path of length \(\left[\frac{m+1}{2}\right] 2^k\). Next in the high level \(k\)-crossed cube the length of the path to cover all nodes that is a Hamiltonian path will be \(2^k\).

Hence the maximum path length that is the diameter of MCC is given by
\[D(MCC) = \left[\frac{m+1}{2}\right] 2^k+ 2^k\]
\[= \left[\frac{m+1}{2}\right] + 1 \right) 2^k.\]

Hence the result.

Cost: The cost of a network is defined as the product of node degree and the network diameter.

Theorem 4.18: The cost of the Meta crossed cube is \((m + k)\left(\frac{m+1}{2}\right) + 1 \right) 2^k\) \(\quad(4.13)\)
Proof: The node degree of Metacrossed cube is \((m+k)\) and the diameter is \(\left\lceil \frac{m+1}{2} \right\rceil + 1 \right\rceil + 1\). Hence the result follows.

**Node Disjoint paths:**

A set of paths is said to be node disjoint if no node other than the source and destination node appears in more than one path. The number of node disjoint paths between any two nodes is an important attribute of any interconnection network. It provides a way of selecting alternative paths in case a path fails and is used for speeding up data transfer between nodes. Thus, the number of such paths provides a measure of the fault tolerance, average delay and reliability of the network.

For the MCC network the node degree is \((m+k)\). So, there exist \((m+k)\) number of node disjoint paths between any two nodes.

**Bisection Width:**

The bisection width of a network is defined as the number of edges whose removal will result in two distinct sub networks. The minimum bisection width is a vital parameter in measuring the area complexity of VLSI layouts of interconnection networks.

**Theorem 4.19:** The bisection width of Metacrossed cube is \(2^{mh-1}\), where \(h=2^k\).

**Proof:** For Meta crossed cube it contains the same number of nodes and edges as that of Metacube. So, the minimum number of edges that will be removed to divide MCC network into two distinct sub networks namely MCC\(0\) (k,m) and MCC\(1\) (k,m) will be the bisection width. The MCC\(0\) (k,m) contains half of the clusters of class \(i\), for \(i=0,1,2,...,2^k-1\) and MCC\(1\) (k,m) contains the rest clusters. In MCC(k,m), there are \(2^{mh+k}\) number of nodes and \(h=2^k\) number of classes.

Hence the Bisection width of the MCC network is \(\left(\frac{2^{mh+k}h}{2^k}\right)/2 = 2^{mh-1}\).

**Average Distance:**

The average inter node distance of a regular cube type network is defined as the ratio of sum of distances between a node and all other nodes to the total number of nodes.

**Theorem 4.20:** The average inter node distance in Meta crossed cube network is given by

\[
\bar{D} = (\bar{d}_{cc}2^k) + 2^k,
\] (4.15)
where \( \bar{d} \) is the average distance in Crossed cube.

**Proof:** In MCC network each cluster is a crossed cube and there are \( 2^k \) such clusters. Let \( \bar{d}_{cc} \) be the average distance in crossed cube. Then the average distance of any two nodes in MCC will be given by

\[
\bar{D} = (\bar{d}2^k) + 2^k
\]

If the basic building block is a \( CC_2 \) then \( \bar{d}_{cc} = 1 \) [51]. So, the average distance is the same as that of MC(k,2). If it is a \( CC_3 \), then \( \bar{d} = \frac{11}{8} \) [51]. Then \( \bar{D} = (\frac{11}{8})2^k + 2^k \).

In general, for Crossed cube of even degree \( \bar{d}_{cc} = \frac{1.18(9m^2+7m)}{1.5m+1} \) (4.16)

And for Crossed cube of odd degree \( \bar{d}_{cc} = \frac{11}{8} + \frac{21}{32}(k - 1) \) (4.17)

So, the average distance of MCC(k,m) can be calculated by using either Equation 4.16 or 4.17 in Equation 4.15 depending upon whether \( m \) is even or odd.

**Message Traffic Density (\( \eta \)):**

The message traffic density of a cube based network is given by \( \eta = \frac{\bar{D}\times p}{E} \), where \( E \) is the total number of links [196]. Assuming, that each node is sending one message to a node at distance \( \bar{D} \) on the average and considering the availability of \( n \) links to accommodate such a traffic, \( \eta \) can be a good measure to estimate the message traffic in the network. The \( p \) is the total number of nodes.

**Theorem 4.21:** The message traffic density of MCC(k,m) is given by \( \eta = \frac{2\bar{D}}{k+m} \) (4.18)

**Proof:** According to the definition \( \eta = \frac{\bar{D}\times p}{E} \).

For MCC(k,m) \( \bar{D} = \bar{d}_{cc}2^k + 2^k \) and total number of nodes is \( p = 2^{mh+k} \).

Total number of edges in MCC is given by \( E = 2^{mh+k} \left( \frac{k+m}{2} \right) \). So the message traffic density for MCC network is given by

\[
\eta = \frac{\bar{D}\times p}{E} = \frac{(\bar{d}2^k+2^k)2^{mh+k}}{2^{mh+k} \left( \frac{k+m}{2} \right)} = \frac{2(\bar{d}^22^k+2^k)}{k+m} = \frac{2\bar{D}}{k+m}.
\]
Mean Internode Distance Rate:

The absolute mean internode distance rate denoted by \( \gamma_a \) and the relative one is denoted by \( \gamma_r \) for any network \( X \) is defined as

\[
\gamma_a (X) = \frac{\bar{d}(x)}{\bar{d}(H^C_n)} \quad \text{and} \quad \gamma_r = \frac{\bar{d}(H^C_n) - \bar{d}(x)}{\bar{d}(H^C_n)}
\]

For MCC network these two parameters are derived as follows:

\[
\gamma_a (MCC) = \frac{d_{2k+2k}}{n2^{n/2}(2^{n-1})}, \quad \text{where} \quad d = \text{average distance of} \ CC_m \text{and} \ n = m2^k + k
\]

and

\[
\gamma_r (MCC) = \frac{n2^n(d_{2k-2k})}{n2^{n/2}(2^{n-1})}
\]

Table 4.2 below shows the comparison of topological properties of MCC network with Hypercube, Crossed cube and Metacube.

<table>
<thead>
<tr>
<th>Network</th>
<th>Degree</th>
<th>Diameter</th>
<th>Cost</th>
<th>Avg. Distance</th>
<th>Bisection width</th>
</tr>
</thead>
<tbody>
<tr>
<td>HC</td>
<td>( n )</td>
<td>( n )</td>
<td>( n^2 )</td>
<td>( n/2 )</td>
<td>( 2^n/2 )</td>
</tr>
<tr>
<td>CC</td>
<td>( n )</td>
<td>( \left\lceil \frac{n + 1}{2} \right\rceil )</td>
<td>( n \left\lceil \frac{n + 1}{2} \right\rceil )</td>
<td>( d )</td>
<td>( 2^n/2 )</td>
</tr>
<tr>
<td>MC</td>
<td>( k + m )</td>
<td>( (k+1)2^k )</td>
<td>( (k + m)(m + 1)2^k )</td>
<td>( \frac{m}{2}2^k + 2^k )</td>
<td>( 2^{2k}/2 )</td>
</tr>
<tr>
<td>MCC</td>
<td>( k + m )</td>
<td>( \left\lceil \frac{m + 1}{2} \right\rceil + 1 ) ( 2^k )</td>
<td>( (k + m) \left\lceil \frac{m + 1}{2} \right\rceil + 1 ) ( 2^k )</td>
<td>( \bar{d}2^k + 2^k )</td>
<td>( 2^{2k}/2 )</td>
</tr>
</tbody>
</table>

4.3.3 Routing and Broadcasting in MCC

In parallel systems, the interconnection networks are employed efficiently to run algorithms whose computations are distributed over the nodes of the network. To run the algorithm always the shortest path among the nodes has to be determined. The MCC network is a very large scale network. So, it is always necessary to find the shortest path available at each step. This section describes routing and broadcasting for Meta crossed cube network and the broadcasting time is compared with the parent networks.

**Routing:** The problem of finding a path from the source node ‘S’ to a destination node ‘T’ and forwarding message along the path is known as the routing problem. Following notations are adopted for MCC(\( k, m \)):
Let $S$ be the source node with node address $(c, m_{h-1}, \ldots, m_0)$.

Then $S$ will have $(m+k)$ links and thus $(m+k)$ number of neighbors. Let $S^i$, $0 \leq i \leq m - 1$ be the $i^{th}$ dimension neighbor in the low level crossed cube. So the node addresses of $S$ and $S^i$ will differ in $i^{th}$ bit position in $m_{c}$, where $c$ is the class address.

Next let $S^i, m \leq i \leq m + k - 1$ be the $i^{th}$ dimension neighbor of $S$ in the high level crossed cube. This means that a node will have $m$ neighbors in the same cluster and $k$ neighbors in the high level crossed cube.

Now let $s$ and $t$ be two nodes in MCC$(k,m)$ with node addresses $(c^s, m^s_{h-1}, \ldots, m^s_0)$ and $(c^t, m^t_{h-1}, \ldots, m^t_0)$ respectively. Next, the problem is to find the path from $s$ to $t$. The routing algorithm is proposed as follows:

*Algorithm:

Procedure MCCRouting

begin

Step 1: Find a path from $c^s$ to $c^t$ in the high level crossed cube. This will be a Hamiltonian path covering all the $2^k$ classes.

Step 2: Now for each class, a shortest path $P_i$ has to be obtained considering only the $m$ bit node address part in the same cluster that is in the $m_i$th address field, for $0 \leq i \leq h-1$.

That is for each node, find the shortest path using crossed cube routing [27].

Step 3: Next all the $P_i$'s are concatenated.

Step 4: Whenever a class change is there, a cross edge is inserted between $P_{i-1}$ and $P_i$ for $1 \leq i \leq h-1$

end

While choosing the cross edge, the crossed cube routing is applied considering the class field addresses bits that is $c_i$'s.

*Broadcasting in MCC Network*

This subsection describes one to all broadcasting in the MCC network. In the network the communication channels are bidirectional.

BroadcastMCC

begin

Step 1: if $k=1$ then double broadcast to a node in the same cluster as well as to the neighboring class
Else Double broadcast in the high level crossed cube to other classes.

Step 2: for $i = 0$ to $2^k-1$

for each cluster in which at least one node has the message

para do

Step 3: crossed cube broadcast to all other nodes in the cluster (either regular or double broadcast)

Step 4: find next node in the cluster using crossed cube routing [27]

endfor

endfor

end

Illustration:

Let $k=1$ and $m=2$. The distance between the nodes 0 and 31 is determined as follows:

$D(0,31)= D(00000,11111)=5$, the hamming distance. Then the distance in Metacube is 5. But in case of Metacrossedcube the distance will be determined with the help of the spanning broadcast tree [11,23] as shown in Fig. 4.9. The height of the broadcast tree for MCC(1,2) is 6. The distance between nodes 0 and 31 is $D(0,31)=4$. But in MC(1,2) network the distance is $D(0,31)= 5$ (the path is 0-1-3-19-23-31).

![Broadcast tree for MCC(1,2) of height 6](image-url)

Figure 4.9: Broadcast tree for MCC(1,2) of height 6
4.3.4 Performance Analysis of MCC

Fault Tolerance:
In the MCC(k, m) network the node connectivity is (m+k)

So, fault tolerance = m+k-1.

Thus, the MCC network can tolerate up to (m+k-1) faults.

Cost effectiveness factor:

Theorem 4.22: The cost effectiveness factor of MCC (m, k) is

\[ \text{CEF} = \frac{1}{1 + \rho \left( \frac{m+k}{2} \right)} \]  

(4.19)

where ‘\( \rho \)’ is the ratio of link cost to processor cost.

Proof: The total no of processors in MCC network is given by

\[ p = 2^{mh+k}, \text{ where, } h = 2^k \]

The total number of edges in MCC network is

\[ E = 2^{mh+k} \left( \frac{m+k}{2} \right) \]

\[ = p \left( \frac{m+k}{2} \right) = f(p). \]

As we know that \( g(p) = \frac{f(p)}{p} \), where \( f(p) = \text{total no of edges} \). Now cost effectiveness factor of MCC network is given by

\[ CEF(p) = \frac{1}{1 + \rho g(p)} \]

For MCC network, \( f(p) = p \left( \frac{k+m}{2} \right) \),

and \( p = 2^{mh+k}, \text{ where } h = 2^k \)

So,

\[ g(p) = \frac{f(p)}{p} = \frac{p \left( \frac{m+k}{2} \right)}{p} = \left( \frac{m+k}{2} \right) \]

Hence,

\[ CEF(p) = \frac{1}{1 + \rho g(p)} = \frac{1}{1 + \rho \left( \frac{m+k}{2} \right)} \]

Time cost effectiveness factor:

Theorem 4.23: The time cost effectiveness factor of the MCC network is given by
Chapter IV Some Studies on Parallel Computer Interconnection Network Topology

\[ TCEF(p, T_p) = \frac{1+\sigma}{1+\rho \left( \frac{m+k}{2} \right) + \frac{\sigma}{p}} \quad (4.20) \]

Proof: In MCC the total number of nodes is

\[ p = 2^{mh+k} \]

From Theorem 4.22,

\[ g(p) = \left( \frac{m+k}{2} \right) \]

For MCC the TCEF is formulated as follows:

\[ TCEF(p, T_p) = \frac{1+\sigma T_r^{a-1}}{1+\rho g(p) + T_r^{a-1} \frac{\sigma}{p}} \quad \text{(For calculation } \alpha \text{ is taken as 1, i.e linear time penalty)} \]

Hence

\[ TCEF(MCC) = \frac{1+\sigma}{1+\rho g(p) + \frac{\sigma}{p}} \]

= \frac{1+\sigma}{1-\rho \left( \frac{m+k}{2} \right) + \frac{\sigma}{p}}

Reliability:

In MCC(k,m), there are \((m+k)\) number of node disjoint paths. First all the possible node disjoint paths in different MCC(k,m)'s are derived and then the two terminal reliability between the two farthest nodes is evaluated. In MCC the lowest possible node degree is three with \(m=2\) and \(k=1\). In degree three MCC the two farthest nodes are 0 and 15 with node addresses (00000 and 001111) and the possible node disjoint paths between them are as follows:

0-16-20-28-12-14-15
0-1-19-27-31-13-15
0-2-3-17-21-29-15

Here number of links is six and number of nodes is five. Hence two terminal reliability for MCC(1,2) is given by

\[ TR(MCC) = 1 - (1 - R_1^2 R_n^5)^3 = 1 - (1 - R_1^2 R_n^{2r-1})^r \]

where \(r = k+m\).

Next for \(k=1\) and \(m=3\), the MCC(1,3), the node connectivity is 4 and thus, there are 4 node disjoint paths between any two nodes. There are 128 nodes in total. So, the node disjoint paths between nodes 0 and 127 are as follows:

0-1-67-91-25-29-95-127 (number of links=8, number of nodes =7)
0-4-5-71-103-127 (number of links=6, number of nodes =5)
Hence, the terminal reliability expression is given by

\[ TR(MCC) = 1 - ((1 - R^2 R^2) (1 - R^6 R^6) (1 - R^7 R^7)) \]
\[ \leq 1 - (1 - R^8 R^8 R^8) (1 - R^7 R^7) \]
\[ = 1 - (1 - R^{r-1} R^{r-1} - 1) (1 - R^{2r-2} R^{2r-2}), \text{ where } r = k+m, \text{ node degree of MCC.} \]

4.4 Proposed Topology: Meta star (M-star)

This section presents the detail construction, addressing or labeling of the nodes of the Meta star graph.

4.4.1 Construction of M-star

This subsection proposes a new interconnection network called the Meta star network suitable for large scale parallel systems. This new network inherits the efficient features of both the Metacube [132,133] as well as the star graph [12]. Like MC network it can connect to a large number of nodes with a quite small node degree as compared to that of MC network. Here each cluster is a star graph of dimension 3 or more. The Meta star network contains \( 2^k \) classes and each class contains \( m! \) number of clusters and each cluster in turn contains \( m! \) number of nodes, as each cluster is a star graph. The Meta star of dimensions 3 and 4 are shown in Fig. 4.13 and 4.14.

Addressing the nodes of Meta star:

The address of a node in the Meta star network has three parts as shown below in Fig. 4.10. The leftmost k-bits in binary represent the class address. Next m bit permutation represents the cluster and the right most m bit permutation represents the node within the cluster. Alternatively decimal notation can be used to reduce the complexity in the diagram. So the nodes in 3-star can have labels 1,2,3,4,5 and 6 as shown in Fig. 4.11. Figure 4.12 shows two individual clusters belonging to class 0 and class 1 of Meta star(1,3) with their node addresses in equivalent decimal notations.

Neighbours:

Given a node \( v(x,y,z) \) of Metstar(k,m), x is the k bit class label, y is the m bit permutation for cluster label and z is the m bit permutation for node label. So, v can have following nodes as neighbour: \( (x_{k-1}x_{k-2}...x_1,x_0,y_1y_2...y_m,z_1z_2...,z_m) \) when \( y_i = z_i \) for all values of i,
Chapter IV  

Some Studies on Parallel Computer Interconnection Network Topology

(k-bit class, \(123...m\), \(123...m\))

class address  cluster address  node address

Figure 4.10: Format of a node address for the Meta star network

![Diagram of Meta star network](image)

Figure 4.11: Node address of 3-Star, (a) 3 permutation digits, (b) one decimal digit

\((x_{k-1}x_{k-2}...x_0, y_1y_2...y_m, z_1z_2...z_m)\) and \((x_{k-1}x_{k-2}...x_0, y_1y_2...y_m, z_1z_2...z_m)\). The third type neighbour belongs to the same cluster. As per the Metacube properties, two nodes belonging to same class but different cluster are not connected. So, to calculate the distance between such nodes two extra edges will be needed one for entering the other class and one for leaving the class. When the nodes belong to same cluster the distance is calculated as per the star graph terminology depending upon the permutations in the node address.

**Illustration:**

In case of Meta star(1,3), \(k = 1, m = 3\). So each node will have \((k+m-1)\) neighbours out of which \(k\) neighbours will be in other class and \((m-1)\) neighbours in the same cluster. Meta star(1,3) contains 2 classes and each class contains 6 clusters. Each cluster in turn contains 6 nodes. Thus there are 36 nodes in each class and in total there are 72 nodes with node degree 3. So for node \((0,123,123)\) the neighbours will be \((1,123,123), (0,123, 213)\) and \((0,123,321)\). In alternate notations as discussed above, the node \((0,1,1)\) will have neighbours \((1,1,1), (0,1,2)\) and \((0,1,6)\) as

137
shown in Fig. 4.12. Out of these three neighbours, the first one belongs to the other class and the rest two belong to the same cluster of the same class.

Neighbours in Meta star(1,3): The following nodes are the neighbours through cross links in the Meta star(1,3):

(a) (0,1,1)-(1,1,1) (0,2,1)-(1,1,2) (0,3,1)-(1,1,3) (0,4,1)-(1,1,4) (0,5,1)-(1,1,5) (0,6,1)-(1,1,6)
(b) (0,1,2)-(1,2,1) (0,2,2)-(1,2,2) (0,3,2)-(1,2,3) (0,4,2)-(1,2,4) (0,5,2)-(1,2,5) (0,6,2)-(1,2,6)
(c) (0,1,3)-(1,3,1) (0,2,3)-(1,3,2) (0,3,3)-(1,3,3) (0,4,3)-(1,3,4) (0,5,3)-(1,3,5) (0,6,3)-(1,3,6)
(d) (0,1,4)-(1,4,1) (0,2,4)-(1,4,2) (0,3,4)-(1,4,3) (0,4,4)-(1,4,4) (0,5,4)-(1,4,5) (0,6,4)-(1,4,6)
(e) (0,1,5)-(1,5,1) (0,2,5)-(1,5,2) (0,3,5)-(1,5,3) (0,4,5)-(1,5,4) (0,5,5)-(1,5,5) (0,6,5)-(1,5,6)
(f) (0,1,6)-(1,6,1) (0,2,6)-(1,6,2) (0,3,6)-(1,6,3) (0,4,6)-(1,6,4) (0,5,6)-(1,6,5) (0,6,6)-(1,6,6)

Figure 4.13: Meta star network of dimension 2, M-star(1,2)
4.3.2. Topological Properties of M-star

This section highlights some of the topological properties of the proposed Meta star network.

**Nodes:**

**Theorem 4.24:** The total number of nodes in the Meta star network is given by

\[ p = 2^k m! m! \]  \hspace{1cm} (4.21)

**Proof:** The node address in the Meta star contains three parts namely class address, cluster address and node address. The Meta star\((k,m)\) network contains \(2^k\) classes. Each class contains \(m!\) number of clusters. Each cluster contains \(m!\) number of nodes. Hence, the total number of nodes is given by

\[ p = 2^k m! m! \]

![Figure 4.14: Meta star network of dimension 3, M-star(1,3)](image)
Node degree:

**Theorem 4.25:** The node degree of the Meta star network is \((k+m-1)\).

\( (4.22) \)

**Proof:** In the Meta star network each cluster is an \(m\)-Star. So each node in the cluster is having \((m-1)\) neighbours. Again the nodes of one cluster in a class are connected to \(k\) nodes in clusters of other classes. The nodes that belong to the clusters of same class are not connected. Hence, for a single node there will be \((k+m-1)\) neighbors. Hence the node degree of the Meta star network is \((k+m-1)\).

Number of links (edges):

**Theorem 4.26:** The total number of edges in the Meta star network is given by

\[
\begin{align*}
E &= \frac{(2km!m!)(k+m-1)}{2} \\
\end{align*}
\]

\( (4.23) \)

**Proof:** In the Meta star network the node degree is \((k+m-1)\). The total number of nodes is \(p=2^k m! m!\). For cube based networks the number of edges is given by

\[
\begin{align*}
E &= \text{(Number of nodes)} \times \text{(Node Degree}/2).
\end{align*}
\]

So for Meta star network the number of edges is given by

\[
E = \frac{(2^k m! m!)(k+m-1)}{2}
\]

Illustration:

Let \(k=1\) and \(m=3\) then, in the Meta star\((1,3)\) there are \(2^1=2\) classes. Each class contains \(m!=6\) clusters and \(m!=6\) nodes in each cluster that is a \(3\)-Star. So the total number of nodes is given by

\[P=2^1 \times 3! \times 3! = 72\]

and the total number of edges is given by

\[E=(2^k m! m!)/(k+m-1)/2=72 \times 3/2=108\]

\((0,1,1)\) \((0,1,2)\) \((0,1,3)\) \((0,1,4)\) \((0,1,5)\) \((0,1,6)\) are the addresses of 6 nodes of cluster 1 of class 0 as shown in Fig. 4.12.

Diameter:

The diameter is the maximum of the shortest distance between any two nodes of a network over all pairs of nodes.

**Theorem 4.27:** The diameter of the Meta star network is given by

\[
D(G) = 2^k \left( 1 + \left\lfloor \frac{3(m-1)}{2} \right\rfloor \right)
\]

\( (4.24) \)

**Proof:** In Meta star the higher level structure is a cube where as the lower level structure is an \(m\)-Star. In the higher level there are \(2^k\) number of classes. The maximum distance will be decided
using two facts, namely (i) number of hops within the cluster and (ii) number of hops between the classes. The length of the path covering the high level cube is $2^k$. Next travelling within the cluster requires $\left\lceil \frac{3(m-1)}{2} \right\rceil$ steps as it is a star graph. So the diameter of the Meta star network is given by

$$D(G) = 2^k + \left( \left\lceil \frac{3(m-2)}{2} \right\rceil \right) 2^k$$

$$= 2^k \left( 1 + \left\lceil \frac{3(m-1)}{2} \right\rceil \right)$$

Hence the result.

**Cost:**

**Theorem 4.28:** The cost of the Meta star network is given by

$$\xi = \left( \left\lceil \frac{3(m-1)}{2} \right\rceil + 1 \right) 2^k (k + m - 1)$$

(4.25)

**Proof:** From Theorem 4.25, the node degree of the Meta star network is $(k+m-1)$. From Theorem 4.27, the diameter of the Meta star is $\left( \left\lceil \frac{3(m-1)}{2} \right\rceil + 1 \right) 2^k$. As the cost of a network is the product of degree and diameter, hence the cost of the Meta star network is given by

$$\xi = \text{diameter} \times \text{degree}$$

$$= \left( \left\lceil \frac{3(m-1)}{2} \right\rceil + 1 \right) \times 2^k \times (k + m - 1)$$

**Bisection Width**

The bisection width of a parallel interconnection network topology is defined as the total number of edges whose removal will result in two distinct sub networks. The minimum bisection width plays a vital role in measuring the area complexity of VLSI layouts of the network topology.

**Theorem 4.29:** The bisection width of the Meta star network is $(m!)^2 \times 2^{k-2}$. 

(4.26)

**Proof:** The total number of nodes in the Meta star network is given by Equ. 4.21. So,

$$p = 2^k m! \ m!$$

There are $2^k$ number of classes in total. After bisection the network will be divided into two equal halves. Suppose $M-star^0(k,m)$ and $M-star^1(k,m)$ such that $M-star^0(k,m)$ will contain half of the clusters of class $i$, $i = 1, 2, \ldots 2^k - 1$ and $M-star^1(k,m)$ will contain the rest clusters. Hence, the bisection width of the $M$-star$(k,m)$ network is
Chapter IV Some Studies on Parallel Computer Interconnection Network Topology

Average Node Distance:

**Theorem 4.30:** The average node distance of the Meta star network is \( D = (\bar{d}_s + 1)2^k \)  
where \( \bar{d}_s \) is the average node distance of the Star graph.

**Proof:** The Meta star is a hybrid network and contains n-Star as a cluster. So, for moving within the cluster the average distance will be same as that of the star graph that is

\[ \bar{d}_s = n - 4 + \sum_{i=1}^{n} \frac{1}{i} \]

Next in the higher level, routing algorithms will traverse to a cluster of each class and there are \( 2^k \) classes hence, the average distance of any two nodes in the M-star\((k,m)\) network is

\[ \bar{D} = \bar{d}_s2^k + 2^k \]

Message Traffic Density:

**Theorem 4.31:** The message traffic density of the Meta star network is given by

\[ \eta = \frac{2\bar{D}}{(k+m-1)} \]  
(4.28)

**Proof:** The total number of nodes in the Meta star network is given by \( 2^k m! m! \) and the total number of edges is \((2^k m! m!)(k+m-1)/2\).

Now using Theorem 4.30,

\[ \bar{D} = \bar{d}_s2^k + 2^k \]

So

\[ \eta = \frac{2^k m! m! \bar{D}}{2^{k+1} m! (k+m-1)/2} \]

Node disjoint paths:

**Theorem 4.32:** The number of node disjoint paths between any two nodes of M-star\((k,m)\) network is \((k+m-1)\).

**Proof:** In the M-star network the lower level network is a Star graph whose node connectivity is \((m-1)\). For the higher level cube the node connectivity is simply \(k\). So the node connectivity of the
M-star \((k,m)\) network is \((k+m-1)\). Hence the number of node disjoint paths between any two nodes of M-star network is \((k+m-1)\).

### 4.4.3 Routing and Broadcasting in M-star

In this section the routing and broadcasting algorithms for the proposed Meta star network are proposed. The Hamiltonian properties are discussed in the beginning.

**Hamiltonian Properties of Meta star:**

The Metacube is Hamiltonian [131,134]. Next Star is also Hamiltonian as shown in Fig. 4.12 and 4.13. For 3-Star the Hamiltonian path will be 1-2-3-4-5-6-1 or 1-6-5-4-3-2-1. The path length is \((n!-1) = 5\). For 4-Star the path will be as follows:

Cluster\(_1\)(1-2-3-4-5)-Cluster\(_2\)(5-4-3-2-1-6)-Cluster\(_3\)(5-4-3-2-1-6)-Cluster\(_4\)(6-1-2-3-4-5)-Cluster\(_1\)(6). Hence, the path length= \((n!-1) = 23\).

Hence n-Star is also Hamiltonian. Meta star being a hybrid network will be also Hamiltonian. The M-star(1,2) is shown to be Hamiltonian in Fig. 4.15.

**Routing in Meta star**

Suppose \(s\) and \(t\) be two nodes in the Meta star\((k,m)\) network. The nodes \(s\) and \(t\) belong to classes \(C^s\) and \(C^t\) respectively, the source class and destination class. Then there will be three cases.

Case (i) : \(s\) and \(t\) both belong to the same class and same cluster.

Case (ii) : \(s\) and \(t\) both belong to the same class but different clusters.

Case(iii) : \(s\) and \(t\) both belong to different class.
Chapter IV

For first case routing will be same as n-Star routing. For second and third case following is the algorithm.

**Algorithm M-star Routing(k,m,s,t)**

```{ 
Step 1: Construct a Hamiltonian path from source class C_s to destination class C_t.
Step 2: In each class, a shortest path P_i will be decided using star routing within each cluster.
Step 3: Each P_i will be concatenated to form the path from s to t.
}
```

Whenever there is a change of class a cross link will be inserted according to the Hamiltonian path found in step 1.

**Illustration:**

Let s and t be two farthest nodes in the Meta star(1,3) network. The node addresses of s be (0,1,1) and t be (0,4,22). Then the path from s to t is as follows:

(0,1,1)*****(1,1,1)—(1,1,7)—(1,1,13)—(1,1,19)******(0,4,19)—(0,4,20)—(0,4,21)—(0,4,22),

(*** represents cross link and --- represents star link that is link within cluster). So distance is 8.

**Broadcasting**

The broadcasting algorithm for the Meta star network is proposed as follows.

**Algorithm: M-starBroadcast(k,m)**

```{ 
Step 1: Broadcast the message from s to all other clusters.
Step 2: For i=0 to 2^k — 1 do
    For each cluster in which at least one node has the message para do
    Step 3: Broadcast the message to all other nodes in the cluster
    Step 4: Find the next neighbour in the cluster using star routing.
    endfor
endfor
}
```
4.4.4 Performance Analysis in M-star

Cost Effectiveness Factor:

**Theorem 4.33:** The cost effectiveness factor of Meta star network is given by

$$\text{CEF}(p) = \frac{2}{2+p(k+m-1)} \quad (4.30)$$

*Proof:* The total number of nodes in the M-star network is given by $p = 2^k m! m!$
and the total number of links in the network is $E = (2^k m! m!)(k+m-1)/2$.
Now
$$E = (2^k m! m!)(k+m-1)/2 = f(p).$$
So
$$g(p) = \frac{f(p)}{p} = \frac{p(k+m-1)/2}{p} = \frac{(k+m-1)}{2}$$
Hence,
$$\text{CEF}(p) = \frac{1}{1+p g(p)} \left(1+p \frac{k+m-1}{2}\right) = \frac{2}{2+p(k+m-1)}$$

Time Cost-effectiveness Factor

**Theorem 4.34:** The Time-cost-effectiveness factor of the M-star(k,m) network is given by

$$\text{TCEF}(p, T_p) = \frac{(1+\sigma)2p}{2+2p+\rho p(k+m-1)} \quad (4.31)$$

*Proof:* The total number of nodes in the M-star network is $p = 2^k m! m!$
and the total number of links in the network is $(2^k m! m!)(k+m-1)/2$.
Therefore the TCEF for M-star is formulated as follows:

$$\text{TCEF}(p, T_p) = \frac{1+\sigma T_1^{p-1}}{1+p g(p)+T_1^{p-1} \frac{\sigma}{p}}$$

where $T_1$ is the time required to solve the problem by a single processor using a fastest sequential algorithm. $T_p$ is the time required to solve the problem by a parallel algorithm using a multiprocessor system having $p$ processors and $\rho$ is the cost of penalty / cost of processors. For linear time penalty $\alpha$ is taken as 1.

From Theorem 4.33, $g(p) = \frac{(k+m-1)}{2}$. Hence the TCEF for M-star is given by

$$\text{TCEF}(p, \text{M-star}) = \frac{1+\sigma}{1+p \frac{(k+m-1)}{2} + \frac{1}{p} \frac{1+\sigma}{2+2p+\rho p(k+m-1)}}$$

145
Reliability:
Here the two terminal reliability evaluations are done for different values of $k$ and $m$.
Reliability of M-star$(1,3)$, $k=1$ and $m=3$.

In the M-star$(1,3)$, the total number of nodes is $2^1 \times 3! \times 3! = 72$ nodes. Suppose the two nodes considered are the farthest nodes then their addresses are $(0,1,1)$ and $(0,6,4)$ because the $(0,1,1)$ is the first node of cluster 1 of class 0 and $(0,6,4)$ is the fourth node of cluster 6 of class 0. In a star network the first node and the fourth node are the farthest node as shown in Fig. 4.11 (b). Similarly in Metacube the first cluster and the last cluster of same class are the farthest as there is no direct link between the clusters of same class.

Now all the possible node disjoint paths between these two nodes is as follows:

011-064
011- 111- 116- 061 -062 -063-064
011- 012- 013- 014 -141 -146 - 064
011- 016- 161- 166 - 066 - 065-064

In all the three possible node disjoint paths the number of links is 6 that is $2m$ and number of nodes is 5 that is $(2m-1)$.

Hence, according to Theorem 2.10, the two terminal reliability is given by

$$TR = 1 - \prod_{i=1}^{k+m-1}(1 - R_i^2mR_{2m}^{2m-1})$$

$$= 1 - (1 - R_i^2mR_{2m}^{2m-1})^3$$

For $k = 2, m = 3$, there are four clusters with address $(00, 01, 10, 11)$ or $(0, 1, 2, 3)$ equivalent address in decimal notation. Now in M-star$(2,3)$, the farthest nodes are $(00,1,1)$ and $(00,6,4)$ or $(0,1,1)$ and $(0,6,4)$. All the clusters of M-star$(2,3)$ are shown in Fig. 4.16. There are $(2+3-1) = 4$ node disjoint paths possible. Hence the paths between the above two nodes are as follows:

011- 012- 121- 126- 062-063-064
011- 016- 161- 166 - 066 - 065-064
011- 211-212-213-214-341-342-343-344-244-245-246-064

Hence, for Meta star$(2,3)$ the terminal reliability can be derived as

$$TR(M\text{-star})=1-(1-R_i^{km}R_n^{km-1})^2(1-R_{1}^{km+1}R_n^{km})(1-R_i^{2km}R_n^{2km-1})$$

$$\leq 1-(1-R_i^{2km}R_n^{2km-1})^{k+m-1}$$
4.5 Results and Discussions

This section is devoted towards evaluation and comparison of results of Star crossed cube (SCC), Meta crossed cube (MCC) and Meta star (M-star).

4.5.1 Discussions on SCC

The Star graph has a major drawback that its scalability is poor. In this regard to establish the superiority of the SCC network a comparison is made for the packing density of different networks. The Fig. 4.17 compares the network size against the node degree of the SCC with that of Star, Hierarchical star (HS), Hyperstar, HC and the CC. As compared to the Star, Hierarchical star and the Hyperstar, the growth of Starcrossed cube is slower as shown in Fig. 4.17. But it has better packing density than the crossed cube and the Hypercube with the same node degree. Hence, the SCC has the convenience to design a network of desired size. The degree of a network represents the connectivity and in turn determines the hardware complexity of the network. The node degree of the Starcube and the Starcrossed cube is same but it is higher than that of the Star graph and the traditional hypercube as well as crossed cube due to the hybrid structure. The hierarchical star has degree equal to that of HC and CC. The node degree of hyperstar is equal to that of star graph for even dimension. The comparison of node degree versus network dimension is shown in Fig. 4.18. Thus, the Figure contains only three curves.
In the design of an interconnection topology it is always required to have small diameter as it greatly impacts the message traffic density. While comparing the diameter against the network, it is observed that the Hierarchical star possesses the highest value as its size grows at a very faster rate than the other networks. Hypercube and the crossed cube both possess smaller values as they are smaller networks. The Starcrossed cube stands lowest among the hybrid networks as shown in Fig. 4.19.

The performance of a network is assumed to be better if it has a low cost feature. Hence, the comparison of cost against the node degree is done in Fig. 4.20. The Starcrossed cube being a hybrid network, its cost is more or less equal to that of its parent networks namely: the Hypercube, Crossed cube and the Star graph while containing more nodes as shown in Table 4.3. But it is found to be less than that of the Starcube, the Hierarchical star and the Hyperstar. At the networks are scaled up, especially for link complexity 5 and beyond that this difference becomes more prominent.
Figure 4.18: Comparison of node degree Vs. dimension of SCC

Figure 4.19: Comparison of diameter of SCC
Chapter IV  
Some Studies on Parallel Computer Interconnection Network Topology

![Graph showing comparison of cost of SCC networks](image)

Figure 4.20: Comparison of Cost of SCC

Table 4.3: Comparison of Cost of SCQ with parent networks

<table>
<thead>
<tr>
<th>Degree</th>
<th>Star</th>
<th>SC</th>
<th>SCQ</th>
<th>HC</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>6</td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td></td>
<td></td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>24</td>
<td>20</td>
<td>20</td>
<td>25</td>
</tr>
<tr>
<td>6</td>
<td>35</td>
<td>42</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>7</td>
<td>54</td>
<td>63</td>
<td>56</td>
<td>49</td>
</tr>
<tr>
<td>8</td>
<td>70</td>
<td>80</td>
<td>72</td>
<td>64</td>
</tr>
<tr>
<td>9</td>
<td>96</td>
<td>108</td>
<td>99</td>
<td>81</td>
</tr>
</tbody>
</table>

The average node distance is an important parameter which influences the overall message traffic in the network. Comparison of average node distance of the Starcrossed cube prevails that it is a better network topology at higher dimensions than the other networks. The variation of average distance with respect to dimension is depicted in Fig. 4.21. At higher
dimensions for even values of ‘m’, the SCQ network exhibits better reduction than the counterparts.

Figure 4.21: Average node distance comparison for SCC

Figure 4.22: Message density comparison for SCC
Next the average message density of the proposed topology is compared with the parent networks. The message traffic density of the Hypercube is always 1. Though the crossed cube is closer to this value, the even order crossed cube shows more reduction. For the Star graph the value is higher with more number of nodes at same dimensions. Being a hybrid graph the Starcrossed cube possesses better values with higher packing density. It is always less than that of the Starcube and the Star graph as shown in Fig. 4.22. The values are evaluated for different values of dimension.

![Figure 4.23: Comparison of cost effectiveness factor of SCC](image)

Figures 4.23 and 4.24 above show the variation of cost effectiveness and time cost effectiveness factors with respect to the dimension. The proposed network is cost effective due to the monotonic decreasing nature of the curves similar to hypercube. It is more beneficial when dimension lies between 4 to 8.

The two terminal reliability of the SCC network is evaluated and compared with that of the Starcube network. For comparison different values of n and m are considered. The Fig. 4.25 (a), (b), (c) depict the above comparison with dimension. In all the cases link and node failure rates are assumed to be 0.0001 and 0.001 respectively.
In Fig. 4.25 (a), on a 3-star various sizes of crossed cubes are considered for evaluation of reliability. The evaluations are done at mission time 1000Hrs. Next in the Fig. 4.25 (b) and (c), 4 and 5 dimensional star graphs are considered with crossed cubes of different orders for evaluation. For comparison equal sized Starcube is considered. The comparison reveals that at all values of dimension, the SCC network is more reliable and it maintains the same status. However, reliability of the SC network reduces much faster with increase in dimension.
Chapter IV

Some Studies on Parallel Computer Interconnection Network Topology

Reliability

5 6 7

Dimension

(a)

(b)
4.5.2 Discussions on MCC

Next the different results obtained for the MCC network are discussed. Here the different topological properties such as degree, diameter, average distance and message traffic density are evaluated for MCC network. Next they are comparatively analyzed with that of Metacube and Crossed cube and Hypercube networks of equal size.

The comparison of degree versus the network dimension is shown in Fig. 4.26. Hypercube and crossed cubes have same node degree. Also the Meta cube and Meta crossed cube have the same node degree. The MC and the MCC networks are capable of packing more number of nodes at the same node degree as compared to HC and CC. Thus, in the Figure there are only two curves shown.

The diameter of MCC network is the minimum among all the networks. Total number of nodes in MCC network is equal to that of MC network, which is too high as compared to Hypercube and Crossed cube network at same node degree. So the comparison of diameter shown in Fig. 4.27 proves the superiority of MCC network over the parent networks.
A network is said to have high performance if it has low cost and meets the requirements with respect to the other factors as discussed in this section. The average cost of Metacrossedcube network against network dimension is computed and compared with that of other networks as shown in Fig. 4.28. It is observed that the MCC network possesses the least cost among all other networks. Hence Metacrossed cube is superior to all other networks.

The variation of $D$ that is the average node distance versus links per node is shown in Fig. 4.29. For higher values of $k$ and $m$ the average distance in MCC network is much reduced than that of MC network. The computed values for different values of $k$ and $m$ are shown in Table 4.4.

In the table $r$ stands for node degree of MC and MCC network where $r=k+m$. The symbol $n$ gives the node degree of HC and CC network. For comparison it is assumed that all the four networks contain equal number of nodes but their node degrees vary.
Figure 4.27: Comparison of diameter of MCC network

Table 4.4: Comparison of average node distance in MCC network

<table>
<thead>
<tr>
<th>$k$</th>
<th>$m$</th>
<th>$r$</th>
<th>$n$</th>
<th>HC</th>
<th>CC</th>
<th>MC</th>
<th>MCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>2.5</td>
<td>2.03</td>
<td>4</td>
<td>4.0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4</td>
<td>7</td>
<td>3.5</td>
<td>2.68</td>
<td>5</td>
<td>4.7</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
<td>9</td>
<td>4.5</td>
<td>3.34</td>
<td>6</td>
<td>5.8</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
<td>11</td>
<td>5.5</td>
<td>4.00</td>
<td>8</td>
<td>8.0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>5</td>
<td>13</td>
<td>6.5</td>
<td>4.65</td>
<td>10</td>
<td>9.4</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>6</td>
<td>15</td>
<td>7.5</td>
<td>5.31</td>
<td>12</td>
<td>11.6</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>5</td>
<td>17</td>
<td>8.5</td>
<td>5.96</td>
<td>16</td>
<td>16.0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>6</td>
<td>19</td>
<td>9.5</td>
<td>6.62</td>
<td>20</td>
<td>18.9</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>7</td>
<td>21</td>
<td>10.5</td>
<td>7.28</td>
<td>24</td>
<td>23.2</td>
</tr>
</tbody>
</table>
Chapter IV  Some Studies on Parallel Computer Interconnection Network Topology

Figure 4.28: Comparison of cost of MCC network

Figure 4.29: Comparison of average node distance of MCC(k,m) with MC(k,m), CC and HC

158
Chapter IV

Some Studies on Parallel Computer Interconnection Network Topology

The Figure 4.30 depicts the variation of message density $\eta$ with the network dimension for the MC and MCC networks. In this case the Crossed cube and Hypercube networks are not considered as the total number of nodes in these two networks is quite less. Only MC and MCC networks are taken into consideration as both of them consist of the same number of nodes, the degrees remaining the same. For Hypercube this measure is 1.

Table 4.5: Message traffic density comparison for MCC

<table>
<thead>
<tr>
<th>$k$</th>
<th>$m$</th>
<th>MC</th>
<th>MCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>2.66</td>
<td>2.66</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2.5</td>
<td>2.37</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>2.4</td>
<td>2.32</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>2.33</td>
<td>2.1</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>2.28</td>
<td>2.14</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>2.25</td>
<td>2.12</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>2.22</td>
<td>2.11</td>
</tr>
</tbody>
</table>

Figure 4.30: Comparison of message traffic density of MCC
In Fig. 4.31 the broadcast time of MCC network is compared with that of the MC network. The broadcast time is evaluated in terms of the number of steps the message travels to reach the destination nodes. The MCC network exhibits quite a good improvement over the MC network while connecting to equal number of nodes. Due to change in link configuration the improvement in broadcast time is achieved.

The results for the mean inter node distance rate \( \gamma_a \) is shown in Table 4.6. The mean inter node distance rate is a factor by which the average node distance of a network grows compared to that of an equal sized hypercube. The values of \( \gamma_a \) for the MCC network decreases for higher values of \( k \) and \( m \). The reduction is found to be better in case of Metacrossed cube as compared to Metacube. The values are evaluated for a constant value of \( k \) with different \( m \).
Table 4.6: Mean internode distance rate evaluation

<table>
<thead>
<tr>
<th>$k$</th>
<th>$m$</th>
<th>$\gamma_a (MCC)$</th>
<th>$\gamma_a (MC)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>1.600</td>
<td>1.600</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1.354</td>
<td>1.428</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1.290</td>
<td>1.333</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>1.145</td>
<td>1.272</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>1.153</td>
<td>1.230</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>1.133</td>
<td>1.222</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>1.0182</td>
<td>1.176</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>0.914</td>
<td>1.157</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>0.959</td>
<td>1.142</td>
</tr>
</tbody>
</table>

Table 4.7: Cost effectiveness factor of MCC($k,m$)

<table>
<thead>
<tr>
<th>$k$</th>
<th>$m$</th>
<th>$\rho=0.1$</th>
<th>$\rho=0.2$</th>
<th>$\rho=0.3$</th>
<th>$\rho=0.4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>0.86</td>
<td>0.76</td>
<td>0.689</td>
<td>0.625</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.83</td>
<td>0.71</td>
<td>0.625</td>
<td>0.555</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.80</td>
<td>0.66</td>
<td>0.571</td>
<td>0.500</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0.83</td>
<td>0.71</td>
<td>0.625</td>
<td>0.555</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.80</td>
<td>0.66</td>
<td>0.571</td>
<td>0.500</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.76</td>
<td>0.62</td>
<td>0.526</td>
<td>0.454</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0.80</td>
<td>0.66</td>
<td>0.571</td>
<td>0.500</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.76</td>
<td>0.62</td>
<td>0.526</td>
<td>0.454</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.74</td>
<td>0.58</td>
<td>0.487</td>
<td>0.416</td>
</tr>
</tbody>
</table>
Table 4.8: Time Cost effectiveness factor of MCC(k, m)

<table>
<thead>
<tr>
<th>k</th>
<th>m</th>
<th>(p=0.1)</th>
<th>(p=0.2)</th>
<th>(p=0.3)</th>
<th>(p=0.4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>1.693</td>
<td>1.502</td>
<td>1.350</td>
<td>1.226</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1.655</td>
<td>1.420</td>
<td>1.243</td>
<td>1.106</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1.597</td>
<td>1.331</td>
<td>1.141</td>
<td>0.999</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1.665</td>
<td>1.427</td>
<td>1.249</td>
<td>1.110</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1.599</td>
<td>1.333</td>
<td>1.142</td>
<td>0.999</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1.538</td>
<td>1.249</td>
<td>1.052</td>
<td>0.909</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1.599</td>
<td>1.333</td>
<td>1.142</td>
<td>0.999</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1.538</td>
<td>1.250</td>
<td>1.052</td>
<td>0.909</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1.481</td>
<td>1.176</td>
<td>0.975</td>
<td>0.833</td>
</tr>
</tbody>
</table>

Figure 4.32: Cost effectiveness factor of MCC network
Chapter IV  Some Studies on Parallel Computer Interconnection Network Topology

The variation of cost effectiveness factor for MCC network is shown in Fig. 4.32. The MCC network is more cost effective with increase in dimension. The Tables 4.7 and 4.8 show the computed values for CEF and TCEF of MCC network. The values are listed for different values of $k$ and $m$ and $p$. The variation of TCEF against dimension is demonstrated in Fig. 4.33. With linear time penalty $p$ is assumed to be 0.1. The TCEF attains is maximum for dimension within 4 to 6.

![TCEF vs Dimension](image)

Figure 4.33: Time cost effectiveness factor of MCC network

Normally for reliability evaluation there are three possible cases. Namely i) only paths are reliable, ii) only nodes are reliable, iii) both nodes and paths are unreliable. For the current work, case (iii) is considered for evaluation. The first reliability evaluation is done for MCC and MC network against dimension. Keeping $k$ constant different values of $m$ is considered and Fig. 4.34 (a), (b) and (c) show the variation of terminal reliability with dimension. For the MCC network the reliability is improved at higher dimensions too. For evaluation the link and node failure rate is assumed to be 0.0001 and 0.001 respectively and mission time is 1000 Hrs. When
the value of $k$ is assumed to be 1, the reliability values for the MCC lies within 0.8 to 0.35. At lower dimensions the reliability of MCC is higher than that of the MC network.

Similarly for $k = 2$ the reliability of the MCC lies within 0.7 to 0.35. For higher values of ‘m’ both the networks tend to be equally reliable as shown in Fig. 4.34 (b). The Fig. 4.34 (c) demonstrates the evaluation with $k = 3$. Similar result is obtained for all values of m. Thus, in general MCC is found to have better reliability than the Meta cube network. This improvement is achieved due to change in the link configuration of MCC network.
Figure 4.34: Reliability comparison of MCC with dimension, (a) k=1, m=2,3,4 (b) k=2, m=2,3,4 (c) k=3, m=2,3,4,5.

Figure 4.35: Reliability comparison of MCC and MC network with mission time
Next the two terminal reliability evaluation is done keeping $r$ fixed at 3 ($k=1$ and $m=2$) with different values of mission time. The MCC network is more reliable than the MC network. The comparison is shown in Fig. 4.35.

### 4.5.3 Discussions on M-star

To establish the superiority of the proposed network the comparison of various topological properties of the Meta star network with the contemporary networks are done in the next paragraphs.

The comparison of node degree with respect to dimension is shown in Fig. 4.36. The n-Star has the lowest degree with less number of nodes. The other three hybrid networks the Metacube, Starcube and the Meta star networks contain more number of nodes with same node degree. But the Meta star network exhibits comparatively better values. However, the curve for HS network is not a continuous one, as has only two values within the taken range of dimension. This is due to the reason that its parameter ‘n’ can take only two values that is (2,2) and (3,3).

![Node degree comparison for M-star network](image)

Figure 4.36: Node degree comparison for M-star network
Chapter IV

Some Studies on Parallel Computer Interconnection Network Topology

Dimension (k,m)

Figure 4.37: Comparison of diameter of M-star network

So far as connecting to comparatively large number of nodes the diameter of the Meta star network is more or less equal to that of the Metacube. The Star graph and the Starcube network both possess lower values and they contain less number of nodes. The comparison is shown in Fig. 4.37. The curve for Hierarchical star is a discontinuous one with only two values. The up and down nature of the curve is due to change in values of k.

The cost comparison as shown in Fig.4.38 establishes the superiority of the Meta star network. It possesses low cost property as of Metacube with higher packing density and lower node degree for all values of dimension. The cost of the Starcube network is higher at lower dimensions but it gets reduced with increase in dimension. Thus, the Metastar is a finer alternative with a flexible network size.

The average node distance of various networks is compared with the proposed M-star network. The comparison is shown in Fig.4.39. Since, the M-star can pack a very high number of nodes

167
Chapter IV Some Studies on Parallel Computer Interconnection Network Topology

Figure 4.38: Comparison of cost versus dimension for M-star network

Figure 4.39: Comparison of average distance of M-star network
Chapter IV Some Studies on Parallel Computer Interconnection Network Topology

Table 4.9 shows the total number of nodes or the packing density of all the four networks with respect to node degree. The shown values are computed with \( k = 1 \) and \( m \) taking different values. The Meta star contains maximum nodes at all possible node degrees. Metacube and Hierarchical Star do not exist with degree 2. Similarly Starcube does not contain any value at node degree 2 and 3. Hierarchical folded network (HFN) does not exist at node degree 2, 3 and 4 [207].

Table 4.9: Comparison of packing density for M-star

<table>
<thead>
<tr>
<th>Degree</th>
<th>MC</th>
<th>n-Star</th>
<th>Starcube</th>
<th>Meta star</th>
<th>HS</th>
<th>HFN</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>6</td>
<td>8</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>32</td>
<td>24</td>
<td>72</td>
<td>72</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>4</td>
<td>128</td>
<td>120</td>
<td>24</td>
<td>144,1152</td>
<td>576</td>
<td>576</td>
</tr>
<tr>
<td>5</td>
<td>512</td>
<td>720</td>
<td>96</td>
<td>2304,28800</td>
<td>14400</td>
<td>14400</td>
</tr>
<tr>
<td>6</td>
<td>2048</td>
<td>1520</td>
<td>480</td>
<td>57600</td>
<td>256</td>
<td></td>
</tr>
</tbody>
</table>

The variation of CEF and TCEF with dimension for M-star network is shown in Fig. 4.40 and 4.41 respectively. The evaluations are done for different values of \( m \) keeping \( k \) fixed at 1. The proposed network is highly cost effective as compared to Metacube (as discussed in Chapter 2). The TCEF for M-star attains its maximum when it contains 1152 nodes that is at node degree four.

The terminal reliability of the M-star network is evaluated with respect to dimension. The computed values are compared with that of Meta cube with equal dimension. At all values of dimension the proposed M-star network is more reliable. The comparison is shown in Fig. 4.42 (a) and (b). The evaluations are done with constant link and node failure rate as 0.0001 and 0.001. The Fig. 4.42 (a) shows the comparison for \( k = 1 \) and the variations for \( k = 2 \) are shown in Fig. 4.42 (b).
Chapter IV

Some Studies on Parallel Computer Interconnection Network Topology

Figure 4.40: Variation of cost effectiveness factor of M-star

Figure 4.41: Variation of time cost effectiveness factor of M-star network
Figure 4.42: Comparison of reliability with dimension for M-star network, (a) $k=1, m=2, 3, 4$
(b) $k=2, m=2, 3, 4$
4.6 Conclusions

The current chapter introduced a new class of networks called hybrid networks by combining different seed networks namely Meta cube, Star graph and the Crossed cube. The new networks proposed here are Meta crossed cube (MCC), Star crossed cube (SCC) and the Meta star (M-Star). The different topological properties are derived. The embedding of other networks into the Star crossed cube are also discussed. An upper bound on the maximum number of meshes that can be embedded has been found.

The MCC which is a large scale network provides reduced cost, average node distance so as to have reduced message traffic density. The MCC is also observed to perform better in terms of broadcast time. Based on the results of comparison the Star crossed cube is found to be the better compared to Star, Star cube and the Hierarchical star graph. This is so particularly for variable node size applications with reduced cost, average node distance.

The M-star network is a massively large system with improved diameter, reduced cost and has high packing density. The growth of the Metastar network is found to be very much flexible. The M-star outperforms the star network in scalability.