Almost Regular Robust Graphs for Arbitrary Number of Nodes*

Pradip K Srimani and Sumit Sur
Department of Computer Science
Colorado State University
Ft. Collins, CO 80523
E-mail: srimani@CS.ColoState.Edu

ABSTRACT. Regular graphs play an important role in designing interconnection networks for multiprocessing systems; but these regular graphs like hypercubes or star graphs cannot be constructed with arbitrary number of nodes. The purpose of the present paper is to examine two families of almost regular maximally fault tolerant graphs (based on hypercubes and star graphs respectively) that can be defined for an arbitrary number of nodes.

1 Introduction

The underlying topology of any multiple processor system is, in general, modeled as an undirected graph where the nodes denote the processing elements and the arcs (edges) denote the bidirectional communication channels. Design features for an efficient interconnection topology include properties like low degree, regularity, small diameter, high connectivity, efficient routing algorithms, high fault tolerance, low fault diameter etc. The small diameter helps to keep the interprocessor communication delay low while the low degree of nodes is necessary to limit the number of input-output ports to some acceptable value. The other desirable feature of an interconnection network topology is high fault tolerance or resilience which is normally measured in terms of vertex connectivity of the graph [Har72]; vertex connectivity of a symmetric graph is defined to be the minimum

*This paper is based on the Invited Talk given by the first author in the 8th Mid-West Conference on Combinatorics, Cryptography and Computing, Wichita State University, October 1993.

number of vertices that need be removed to make the remaining graph disconnected and obviously the graph is called maximally fault tolerant when the vertex connectivity is equal to the minimum degree of a node in the graph. Various authors [Pra85a], [Pra85b], [AL82], [Sch91], [LJD93] have investigated the problem of network design with a view to achieving these goals. It has been observed that regular graphs in general and those with strong algebraic structures play the most important role in network design because of the ease of designing uniform routing algorithms as well as of mapping parallel algorithms on the networks. Most popular among these are the well known binary n-cubes or the hypercubes; they have been used to design various commercial multiprocessor machines and they have been extensively studied [Lei90]. A binary hypercube graph H_n is an undirected graph of N $(= 2^n)$ vertices, each representing a distinct n-bit binary number and two nodes are connected by an edge iff the Hamming distance between the two nodes is 1. Figure 1 shows the hypercubes of dimensions 1, 2 and 3. We can easily observe the following:

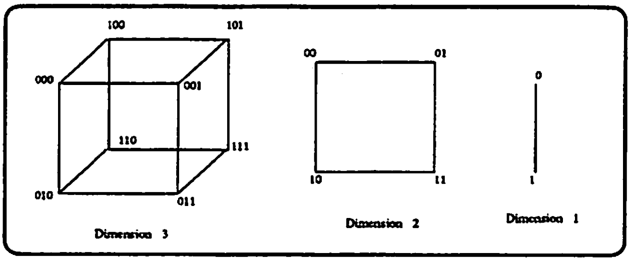

Figure 1: Hypercubes of Dimensions 1,2 and 3

- H_n has $n \times 2^{n-1}$ edges with a diameter of n, i.e. diameter is logarithmic in N and the number of edges is $N\mathcal{F}(N)$ where $\mathcal{F}(N) = O(\log N)$.
- H_n is n-regular with vertex connectivity n, i.e., H_n is **maximally fault tolerant.**
- Fault diameter of H_n is 1+ the fault free diameter.
- H_n can be defined only for integer values of n.

Recently, a new interconnection topology, called the star graphs has been reported in the literature [AK87], [AK89], [SS92a], [DT91]. It is to be noted that these star graphs are a class of Cayley graphs as are n-cubes or the pancake graphs [AK89]. A Star Graph S_n, an undirected Cayley graph of dimension n with $N = n!$ vertices, is defined to be symmetric graph $G = (V, E)$ where V is the set of $n!$ vertices, each representing a distinct permutation of n elements and E is the set of symmetric edges such that two permutations (nodes) are connected by an edge iff one can be reached from the other by interchanging its first symbol with any other symbol. For example in S_3, the node representing permutation ABC will have edges to two other permutations (nodes) BAC and CBA. noted that these star graphs are different from star graphs of [Har72]. These star graphs seem to be very attractive alternatives to the n-cubes in terms of almost all the desirable properties of an interconnection structure. It has been shown that these star graphs can accommodate more processors with less interconnection hardware and less communication delay (compared to n-cubes). Figure 2 shows the star graphs of dimensions 3 and 4.

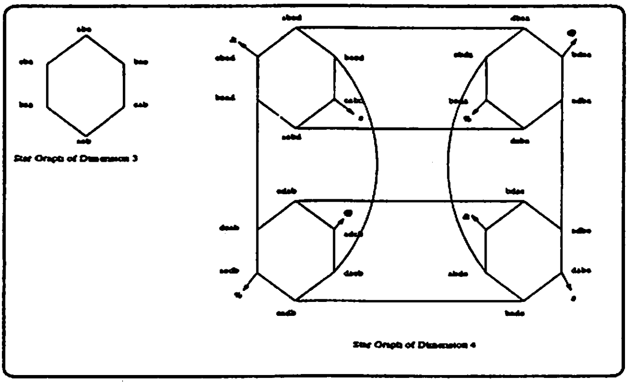

Figure 2: Star Graphs of Dimensions 3 and 4

It can be easily shown that

- S_n has $n!(n-1)/2$ links with a diameter of $\lceil 3(n-1)/2 \rceil$ (i.e., $N\mathcal{F}(N)$ where $\mathcal{F}(N)$ is sublogarithmic).
- S_n is $(n-1)$-regular with a vertex connectivity of $(n-1)$, i.e., S_n is **maximally fault tolerant.**
- S_n can be defined only for integer values of n.
- Fault diameter is $\lceil 3(n-1)/2 \rceil + 1$.

One significant drawback of both of these two topologies is that neither of them is incrementally extensible (expandable). The number of nodes in a hypercube must be some power of two and consequently this topology cannot be defined for an arbitrary number of computing nodes. The problem with the star graphs is even more serious in that the number of nodes in a star graph must be factorial of some integer. This incremental extensibility is a very essential and desirable property in real life applications of a topology in designing computer networks. People have tried over the years to overcome this difficulty especially with the hypercubes. A few generalizations of the hypercube structures have also been proposed [Lei90], [Kat88], [SS88], [BA84], [Sen89] within the last ten years. But, most of the proposed architectures are irregular and the irregularity increases with the size of the graph; for example, in the most recent generalization of the hypercubes, called supercubes [Sen89], the difference between the maximum and the minimum degree of a node can be as high as n and the degrees of the nodes are distributed in the range $(n, 2n)$ for a supercube with N nodes, $2^n < N < 2^{n+1}$. There is only one such study on generalization of star graphs [LB94].

Our purpose in the present paper is to examine two different incrementally extensible network graphs that can be defined for an arbitrary number of nodes: (1) **Incrementally Extensible Hypercubes** or the IEH graphs [SS92a] and (2) **Super Star Graphs** [SS91]. The design philosophy basically involves appropriate interconnection of different sized hypercubes or star graphs of smaller dimensions. More precisely, the proposed families of graphs have the following properties:

- Adding a new node to an existing network is easy and simple; in most cases no reorganization of existing edges is necessary.
- The network is optimally fault tolerant in the sense that the vertex connectivity is equal to the minimum degree of a node in the graph.
- Number of edges is $O(N\mathcal{F}(N))$ where $\mathcal{F}(N)$ is logarithmic or sublogarithmic.
- The diameter is logarithmic or sublogarithmic in the number of nodes.
- The graph is almost regular i.e., the difference between the maximum and the minimum degree of a node is always $\leq c$ (c is a constant independent of the size of the graph; $c = 1$ for IEH graphs and $c = 2$ for the super star graphs).
- Routing and Fault Tolerant Routing should be relatively easy to implement.

2 Incrementally Extensible Hypercubes or IEH Graphs

A hypercube H_n is defined for N vertices only when $N = 2^n$. Our objective is to design a new topology, called *Incrementally Extensible Hypercube* or *IEH* graphs, that can be defined for an arbitrary number of nodes. The topology of our proposed graph consists of an interconnection of an appropriate number of hypercube subgraphs of different sizes. We need a new type of connection besides the usual hypercube connections in a proper hypercube. These edges are needed to connect two hypercubes of different sizes and hence are called *Inter-Cube* or *IC* edges. They are defined as follows.

Definition 2.1: Consider two hypercubes H_i and H_j and without any loss of generality assume $i > j$. Each node in H_i has a i-bit and each node in H_j has a j-bit binary label. Each node $b_{j-1}b_{j-2}\cdots b_0$ in H_j is connected to $(i-j)$ different nodes in H_i: $0\overbrace{1\cdots1}^{i-j-1}b_{j-1}b_{j-2}\cdots b_0$, $0\overbrace{01\cdots1}^{i-j-1}b_{j-1}b_{j-2}\cdots b_0$, $0\overbrace{101\cdots1}^{i-j-1}b_{j-1}b_{j-2}\cdots b_0, \cdots, 0\overbrace{1\cdots10}^{i-j-1}b_{j-1}b_{j-2}\cdots b_0$. These nodes are called the *image* of the node $b_{j-1}b_{j-2}\cdots b_0$ of H_j in H_i.

Figure 3 shows the *IC* edges between H_3 and H_0 and between H_3 and H_2 respectively.

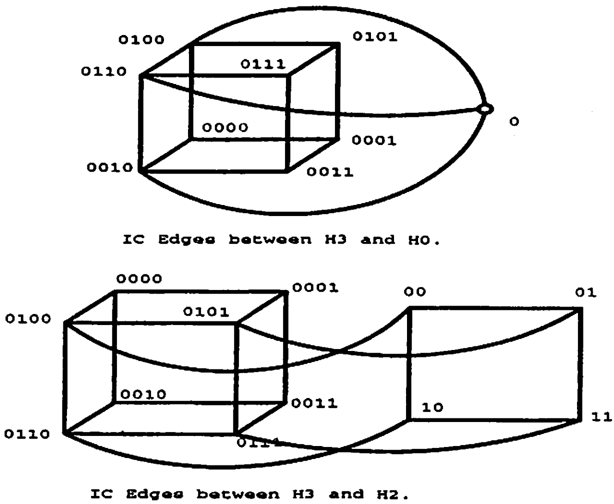

Figure 3: Example IC Edges

Remark: It is to be noted that for any given H_i and H_j, $i > j$, the image sets of any two nodes of H_j are mutually disjoint.

2.1 Construction of IEH Graphs

The basic philosophy in our design of the incrementally extensible hypercube or IEH graphs is to express N as a sum of several powers of 2, i.e., to write N as a binary number, build the smaller hypercubes, and then to add appropriate additional IC edges to connect those smaller hypercubes. The following algorithm builds the incrementable hypercube for any given N, $2^n < N < 2^{n+1}$. If $N = 2^n$ for some n, then we get the proper hypercube.

The Algorithm

Step 1: *[Build the smaller hypercube subgraphs]*

Express N as a $(n+1)$ binary number as $N = < c_n, c_{n-1}, \cdots, c_0 >$, where $c_i \in \{0,1\}$ and $c_n = 1$ since $N > 2^n$. For each c_i, $c_i \neq 0$, construct a hypercube H_i with 2^i nodes.

Step 2: *[Label the nodes]*

Note that each node has a $(n+1)$-bit binary label. Each hypercube H_i is labeled as $\overbrace{11\cdots1}^{n-i}0\underbrace{--\cdots-}_{i}$. Obviously each hypercube of dimension i (having 2^i nodes) has i number of dashes and the individual nodes of the hypercube can be obtained by filling the dashes with 0 or 1 in all possible ways. In other words, the binary representation of each node in H_i has the same prefix of $n-i$ 1's followed by a single zero.

Step 3: *[Construct the incrementable hypercube in steps by providing the extra edges]*

Find the minimum i such that $c_i \neq 0$. Set $j = i$ and $G_j = H_i$. Set $i = i + 1$.
while $i \leq n$ do
if $c_i \neq 0$ then

- if $i - j = 1$ then
  each node x in G_j with label $\overbrace{11\cdots1}^{n-j}b_jb_{j-1}\cdots b_0$ is
  connected to the node $\overbrace{11\cdots1}^{n-j-1}0b_jb_{j-1}\cdots b_0$ of H_i

  else
  each node x in G_j with label $\overbrace{11\cdots1}^{n-j}b_jb_{j-1}\cdots b_0$ is connected to $i-j$ different nodes of H_i chosen in the following way:

$$\overbrace{1\cdots1}^{n-i}0\overbrace{11\cdots1}^{i-j-1}b_jb_{j-1}\cdots b_0$$

$$\overbrace{1\cdots1}^{n-i}0\overbrace{11\cdots10}^{i-j-1}b_jb_{j-1}\cdots b_0$$

$$\overbrace{1\cdots1}^{n-i}0\overbrace{11\cdots01}^{i-j-1}b_jb_{j-1}\cdots b_0$$

$$\cdots$$

$$\overbrace{1\cdots1}^{n-i}0\overbrace{01\cdots1}^{i-j-1}b_jb_{j-1}\cdots b_0$$

- Set $j = i$ and set G_j to be the composite graph generated in the previous steps. Note that G_j has now $\sum_{k=0}^{j} c_k 2^k$ nodes and the binary label of each node in G_j has a prefix of $n-j$ 1's.

$i = i + 1$
Return G_n as the desired incrementable hypercube graph of N vertices.

We can easily observe the following characteristics of the above construction algorithm.

- The resulting incrementable hypercube of N vertices, as designed by the algorithm, is composed of different sized smaller hypercubes which are connected by the *IC* edges as introduced in step 3.
- If $N = 2^n$ for some integer n, then the incrementable hypercube for N vertices is the regular hypercube H_n, i.e., the incrementable hypercubes form a super class of the family of regular hypercubes.
- In step 3, for each i, G_i represents an incrementable hypercube graph of $\sum_{k=0}^{i} c_k 2^k$ vertices.
- In step 3, for each i, whenever *IC* connections are provided between a proper hypercube H_i and some smaller incrementable hypercube G_j, $j < i$, the degree of each node of H_i is increased *at most* by 1. This is evident from three facts: (1) H_i is a hypercube of 2^i nodes, (2) the maximum number of nodes in the incrementable hypercube G_j is $2^{j+1} - 1$, and (3) $2^i > (i-j)\{2^{j+1} - 1\}$ for any integer i and j, $i > j$.
- In step 3, for each i, the Hamming distance between an arbitrary node of G_j and any of its images in G_i is either 1 or 2.

Example 1: Let $N = 11$. Then N can be expressed as a 4-bit binary number 1011, i.e., $c_3 = 1$, $c_2 = 0$, $c_1 = 1$ and $c_0 = 1$. In step 1 we build H_3, H_1 and H_0 separately. In step 2 we label all the nodes with 4-bit binary numbers as shown in Figure 4. In step 3, IC edges are provided from lower order hypercubes to the higher order ones progressively. Note that the single node of H_0 has a single IC edge to H_1; G_1 has 3 nodes; each node of G_1 is connected to two different nodes of H_3. The resulting IEH graph $G_3(11)$ has 20 edges; also, among 11 nodes, 7 has a degree 4 and the rest has a degree 3.

Example 2: Let $N = 13$. N can be expressed as a 4-bit binary number 1101. So we build H_3, H_2 and H_0 and label all the nodes with 4-bit binary numbers in step 2 of the algorithm. And in step 3, IC edges are provided from the lower order hypercubes to the higher order ones progressively. Note that the single node in H_0 has now two IC edges to H_2; G_2 has 5 nodes each of which is connected to a distinct node of H_3 by an IC edge. The resulting IEH graph $G_3(13)$ has 23 edges; 7 of them are IC edges. Out of 13 nodes, 7 has a degree of 4 and the rest of the 6 nodes have each a degree 3.

2.2 Properties of the IEH Graphs

We review different topological properties of the incrementable hypercubes $G_n(N)$, where N is the number of nodes in the graph and the subscript n indicates the fact that $2^n < N < 2^{n+1}$ or that the component hypercube subgraphs are of dimension n or smaller. We use ξ and $\mathcal{D}$ to indicate the vertex connectivity and diameter respectively of any graph. For example, for any hypercube H_n, $\xi(H_n) = n$ and $\mathcal{D}(H_n) = n$. Details of the proofs can be found in [SS92a].

Theorem 2.1. *The vertex connectivity of an IEH graph $G_n(N)$ of N nodes, $2^n \leq N < 2^{n+1}$, is given by $\xi(G_n(N)) = n$.*

Theorem 2.2. *The diameter of $G_n(N)$ is equal to the diameter of the next higher order hypercube H_{n+1}, e.g. $\mathcal{D}(G_n(N)) = \mathcal{D}(H_{n+1}) = n + 1$.*

Theorem 2.3. *Total number of edges in the IEH graph $G_n(N)$, $2^n < N < 2^{n+1}$, where $N =< c_n, \cdots, c_0 >$ in binary, is given by*

$$\sum_{i=1}^{n} c_i i 2^{i-1} + \sum_{i=0}^{n-1} c_i 2^i (n - i) = O(N \log N)$$

Theorem 2.4. *The maximum degree of any node in the IEH graph $G_n(N)$, $2^n < N \leq 2^{n+1}$, is $n + 1$.*

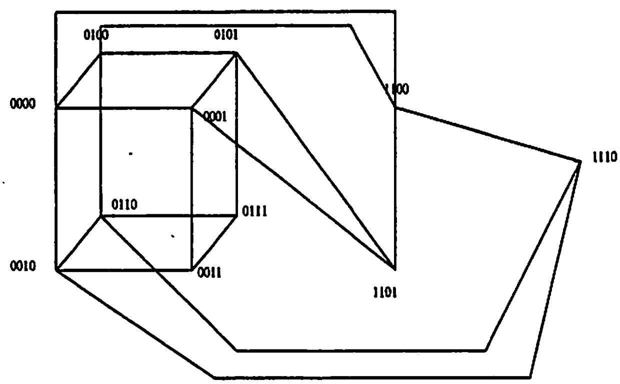

Figure 4: IEH Graph $G_3(11)$ of 11 Nodes

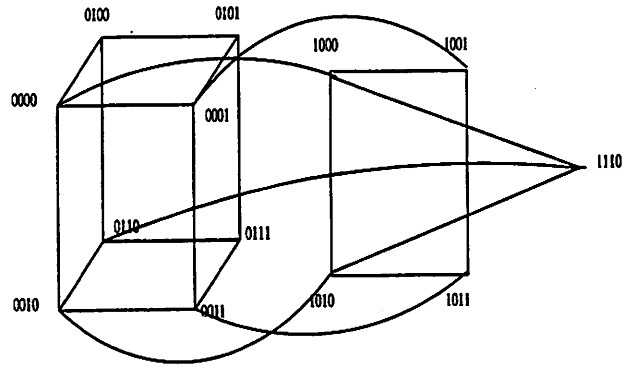

Figure 5: IEH Graph $G_3(13)$ of 13 Nodes

Theorem 2.5. *The number of vertices in $G_n(N)$ with the maximum degree of $n+1$ is given by*

$$\sum_{i=0}^{n-1} c_i 2^i (n-i)$$

Theorem 2.6. *The incrementable hypercube graph $H_n(N)$ is optimally fault tolerant.*

3 Super Star Graphs

In a given star graph S_n, we use V_X to denote the set of nodes (permutations) that end with the symbol X. Obviously, V_X is a star graph of order $n-1$. Similarly, V_α denotes the set of nodes that end with α where α represents a sequence of symbols. V_α is a star graph of dimension $n-|\alpha|$ if V_α is a subgraph of S_n. For example V_{YZ} denotes the set of nodes that end with YZ. Now we introduce several new concepts to facilitate the discussion of our new topology in the next section.

Definition 3.1: Consider any two mutually disjoint subgraphs V_X and V_Y of a star graph S_n. The nodes of V_X, that are directly connected to any node of V_Y, are called the gateway nodes of V_X with respect to V_Y. We denote this set of nodes by $G_{X,Y}$. In general, either or both of X and Y may be sequence of symbols instead of single symbols.

For example, in S_4, $G_{A,B} = \{BDCA, BCDA\}$ and $G_{B,A} = \{ADCB, ACDB\}$ (see Figure 2).

Definition 3.2: Any positive integer N, $n! \leq N < (n+1)!$, can be expressed in its mixed-radix form as $< a_n, a_{n-1}, \cdots, a_1 >$, where

$$N = a_n.n! + a_{n-1}.(n-1)! + ... + a_1.1!$$

and $0 \leq a_i \leq i$ for $i = 1, \cdots, n-1$, and $0 < a_n \leq n$.

For example, $85 =< 3, 2, 0, 1 >$, since $3.4! + 2.3! + 0.2! + 1.1! = 72 + 12 + 1 = 85$.

The topology of our proposed super star graph consists of an interconnection of an appropriate number of star subgraphs of different sizes. We need two types of new connections besides the usual star graph connections in a complete star graph: (a) *type I* connections, induced by the definition of star graphs, and (b) some additional edges, called *type II* connections. We define the type I connection as follows:

Definition 3.3: Given m copies of S_k where $m \leq k$, we say that these m copies are joined by type I connections when the gateway points of each S_k (when viewed as a subgraph of S_n, $n > k$) are connected to their counterparts in each of the other copies of S_k.

Figure 6 shows the type I connections among 3 copies of S_3. Next we need to define the ***group graphs*** or simply the ***groups***.

Definition 3.4: When m copies of S_k, $m \leq k$, are joined by type I connections, the resulting graph is called a group $GR_i(m)$.

Figure 6 shows a $GR_3(3)$. If the m components of a $GR_i(m)$ are numbered from 1 to m as $GR_i^\ell(m)$, $1 \leq \ell \leq m$, the first component or $GR_i^1(m)$ will be called the leader $L_i(m)$ of the group $GR_i(m)$. Note that each component $GR_i^\ell(m)$ of the group is a star graph of dimension i.

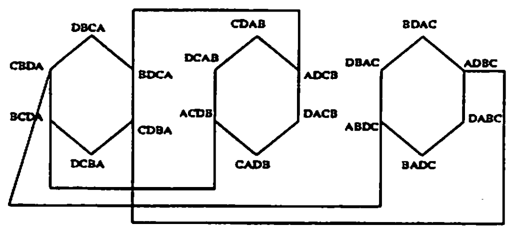

Figure 6: Type I Connections between Star Graphs

Next we need a scheme to number the vertices of a given star graph. It is well known [Knu73] that all the $n!$ permutations of n distinct symbols can be uniquely numbered from 0 through $n! - 1$. We use this scheme to number the vertices of any star graph S_n. We also can extend this scheme to number the vertices of a group $GR_i(m)$. A group $GR_i(m)$ has $m.i!$ vertices; the vertices of GR_i^1 are numbered from 0 to $i! - 1$, the vertices of GR_i^2 are numbered from $i!$ to $2i! - 1$, and so on.

Definition 3.5: Given a subgraph V_α of S_n, where α is an arbitrary sequence of symbols, $|\alpha| < n$, the *head node* is the node in which all symbols other than the fixed ones are in alphabetical ascending order.

For example, the *head node* of V_{CAE}, subgraph of S_7 is the node $BDFGCAE$.

Definition 3.6: Given two star graphs S_m and S_n, $m < n$, S_m and S_n are called to be joined by type II connections if each vertex of S_m is connected to $n-m$ different nodes of S_n by direct edges. The subset of vertices of S_n that are connected to the nodes of S_m is called the type II image of S_m on S_n and has a cardinality of $(n-m)|S_m|$.

Remark: This definition can be easily extended to two arbitrary m-connected and n-connected graphs G and H respectively, provided $|H| \geq (n-m)|G|$, where $|H|$ and $|G|$ denote the number of vertices in H and G respectively.

3.1 Construction of Super Star Graphs

In this section we use the concepts developed in the previous section to design a Super Star graph for an arbitrary number N of given nodes. The basic idea behind our design is to decompose N as a sum of several factorials using mixed-radix representation (when N is not $n!$ for some n), build the smaller star graphs, and then to add appropriate type I and type II edges to connect those smaller star graphs maintaining the desirable properties of high vertex connectivity and low diameter. The following algorithm builds the super star for any given N (we assume $n! < N < (n+1)!$; otherwise the we have original star graph).

The Algorithm

Step 1: *[Build the smaller star subgraphs]*

Compute the mixed radix representation of $N = < c_n, c_{n-1}, \cdots, c_1 >$ and construct c_i copies of S_i for all i, $1 \leq i \leq n$ (note $c_n \neq 0$).

Step 2: *[Label the nodes]*

- Choose $n+1$ symbols to label the nodes (permutations). We use $n+1$ consecutive English letters starting with A.
- For $i = n$ to 1 do the following (fix the i-th symbol for the nodes):

- if $c_i \neq 0$ then label each of the c_i copies of S_i as $V_{\alpha_j\beta}$ where $\beta = symbol(n-i+1)symbol(n-i+2)\cdots symbol(n)$, and α_j, $1 \leq j \leq c_i$, are chosen in alphabetic order from the set of symbols that are yet to be allocated to the "symbol" array.
- Set $symbol(i)$ to be equal to the next available English letter in alphabetic order.

Step 3: *[Provide type I connections among star subgraphs to form groups]*

- For each i, $1 \leq i \leq n$, join the c_i components of S_i's by type I connection as defined earlier to get the different groups GR_i(note that this does not connect the star subgraphs of different sizes).
- Each group GR_i has c_i number of components GR_i^ℓ, $1 \leq \ell \leq c_i$ each of which is a star graph of dimension i. The vertices in GR_i are numbered from 0 to $c_i.i! - 1$ by using the vertex numbering scheme as described before (the vertices of S_i^1 are numbered from 0 to $i! - 1$, those of S_i^2 are numbered from $i!$ to $2i! - 1$ and so on).

Step 4: *[Construct the super star graph in steps by providing the type II connections]*

Find the minimum i such that $c_i \neq 0$ and then set $j = i$ and set $Z_j = GR_i$ (Z_j denotes the super star graph with $\sum_{k=1}^{j} c_k k!$ nodes).
while $i \leq n$ do
if $c_i \neq 0$ then

- Establish type II connections between Z_j and GR_i. Each node in Z_j is connected to $i - j$ different nodes of the leader L_i of GR_i. This is easily done by using the node numberings in both the graphs Z_j and GR_i (e.g., node "0" of Z_j is connected to nodes "0" through node "i-j-1" of L_i, node "1" of Z_j is connected to nodes "i-j" through "2(i-j)-1" of L_i and so on).
- Renumber the nodes of Z_j by adding $c_i i!$ to each node number.
- Set $j = i$ and set Z_j to be the composite graph generated in the previous steps. Note that Z_j has now $\sum_{k=1}^{j} c_k k!$ nodes and they are numbered from 0 to $\sum_{k+1}^{j} c_k k! - 1$.

$i = i + 1$
Return Z_n as the desired super star graph of N vertices.

We can easily observe the following characteristics of the above design algorithm.

- The resulting super star graph of N vertices, as designed by the algorithm, is composed of different sized smaller star graphs which are connected by type I and type II connections.
- If $N = n!$ for some integer n, then the super star graph for N vertices is the original star graph S_n, i.e., the super star graphs form a super class for the family of star graphs.
- In step 4, for each i, Z_i represents a super star graph of $\sum_{k=1}^{i} c_k k!$ vertices.

- In step 4, for each i, whenever type II connections are provided between a leader L_i of a group GR_i and some smaller super star Z_i, $j < i$, the degree of each node of L_i is increased *at most* by 1. This is evident from three facts: (1) a leader L_i is a star graph S_i of $i!$ nodes, (2) the maximum number of nodes in the super star Z_j is $(j+1)! - 1$, and (3) $i! > (i-j)\{(j+1)! - 1\}$ for any integer i and j, $i > j$.

Example 1: Let $N = 19$. Then N can be expressed as $N =< 3, 0, 1 >$ or $c_3 = 3$, $c_2 = 0$ and $c_1 = 1$. Hence there are three groups: GR_3, GR_2 which is null since $c_2 = 0$ and GR_1. GR_3 has 3 components: $GR_3^1 = V_A$ which is also the leader L_3 of this group, $GR_3^2 = V_B$ and $GR_3^3 = V_C$. Each of these components is a star graph of dimension 3. See Figure 7. The vertices of GR_3 are numbered from 0 to $3.3! - 1 = 17$. The numberings are shown in the figure in parenthesis. The group GR_1 has only one component $GR_1^1 = V_{BAD}$ since $symbol(3) = D$ and $symbol(2) = A$ and obviously GR_1^1 is a star graph of dimension 1 i.e., a single vertex $CBAD$. Type I connections are provided in GR_3 by joining all the gateway points to their respective counterparts. Lastly, we need to provide the type II connections between GR_1 and GR_3 and that is done by joining the node 0 of GR_1 to the nodes 0 and 1 of GR_3.

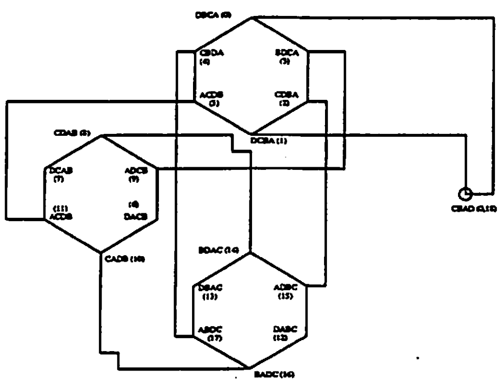

Figure 7: Super Star Graph with $N = 19$

Example 2: Let $N = 23$. Then N can be expressed as $N =< 3, 2, 1 >$ or $c_3 = 3$, $c_2 = 2$ abd $c_1 = 1$. Here there are 3 non-null groups. See Figure 8. There are 3 components of GR_3 and they are numbered from 0 to 17 as before. $symbol(3) = D$ and hence GR_2 has two components V_{AD} and V_{BD} each of which is a star graph of dimension 2; they are numbered from 0 to 3. As before GR_1 is a single node (a star graph of dimension 1) and since $symbol(2) = C$, this is labeled as the permutation $BACD$. Type I connections are provided in each group. Then in step 4 of the algorithm we first provide type II connections to the nodes of GR_1 and GR_2 by joining node "0" of GR_1 to the node "0" of GR_2 and we get Z_2. Nodes of Z_2 are renumbered (actually the nodes of GR_1 need only be renumbered; the node $BACD$ is renumbered as 4). Next, GR_3 and Z_2 are joined by type II connections to get the desired super star graph (nodes "0" through "4" of Z_2 are joined to the nodes "0" through "4" of GR_3).

3.2 Properties of Super Star Graphs

We review different topological properties of the super star graphs $Z_n(N)$, where N is the number of nodes in the graph and the subscript n indicates the fact that $n! < N < (n+1)!$ or that the component star subgraphs are of dimension n or smaller. We use ξ and $\mathcal{D}$ to indicate the vertex connectivity and diameter respectively of any graph. For example, for any star graph S_n, $\xi(S_n) = n - 1$ and $\mathcal{D}(S_n) = \lfloor 3(n-1)/2 \rfloor$. See [SS91] for details of the proofs.

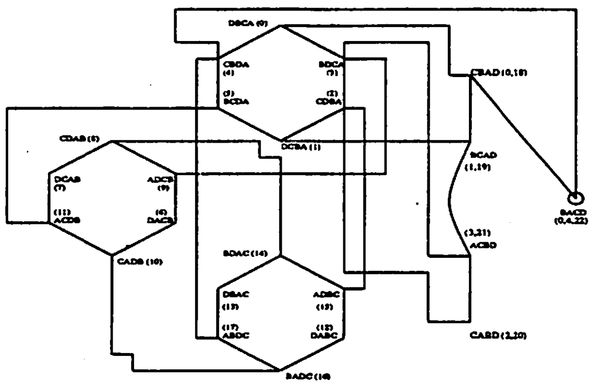

Figure 8: Super Star Graph with $N = 23$

Lemma 3.1. *When m copies of S_n, $m < n$, are connected by type I connections to form a $GR_n(m)$, $\xi(GR_n(m))$ is given by $n - 1$.*

Theorem 3.1. *The vertex connectivity of a super star graph $Z_n(N)$ of N nodes, $n! \leq N < (n+1)!$, is given by $\xi(Z_n(N)) = n - 1$.*

Corollary. *A super star graph of N vertices, where $n! \leq N < (n+1)!$ has the same vertex connectivity as a star graph S_n of $n!$ vertices.*

Lemma 3.2. *For any pair of nodes u and v such that $u \in V_X$ and $v \in V_Y$ in a star graph S_n, there always exists an optimal path between u and v that does not go through any nodes other than those belonging to V_X or V_Y.*

Theorem 3.2. *The diameter of a group $GR_i(m)$ is equal to the diameter of the next higher order star graph S_{i+1}, e.g. $\mathcal{D}(GR_i(m)) = \mathcal{D}(S_{i+1}) = \lfloor 3i/2 \rfloor$.*

Theorem 3.3. *The diameter of the super star graph $Z_n(N)$, $n! < N < (n+1)!$, is given by $\mathcal{D}(Z_n(N)) = \lfloor 3n/2 \rfloor + 1$.*

Theorem 3.4. *Total number of edges in a super star graph $Z_n(N)$, $n! < N < (n+1)!$ and $N =< c_n, \cdots, c_1 >$, is given by*

$$\sum_{i=1}^{n} \left\{ c_i \frac{i!(i-1)}{2} + \binom{c_i}{2}(i-1)! + c_i(n-i)i! \right\}.$$

Theorem 3.5. *The maximum degree of any node in a super star graph $Z_n(N)$ is $n+1$.*

Theorem 3.6. *The upper bound on the number of vertices in $Z_n(N)$ with the maximum degree of $n+1$ is given by*

$$\sum_{i=1,\ c_i \neq 0}^{n} (i-1)!(c_i - 1)$$

Theorem 3.7. *For any super star graph $Z_n(N)$, $n! < N < (n+1)!$, there exists at least one node with degree $n-1$.*

Theorem 3.8. *The super star graph $Z_n(N)$ is optimally fault tolerant.*

4 Conclusion

We have reviewed two families of network graphs for an arbitrary number of computing nodes. Additional nodes can be added to the networks with no or minimal reorganization of the existing interconnection. The topologies have logarithmic or sublogarithmic diameter and is optimally fault tolerant in the sense that the vertex connectivity is equal to the minimum degree of a node. The topology is almost regular, i.e., the difference between maximum and minimum node degree is always ≤ 1. It'd be interesting to study shortest routing as well as fault tolerant routing algorithms for these graphs. Computation of fault diameters for these graphs is also an open problem.

References

[AK87] S.B. Akers and B. Krishnamurthy, The star graph: an attractive alternative to n-cube. In "Proceedings of International Conference on Parallel Processing (ICPP-87), pp. 393–400, St. Charles, Illinois, August 1987.

[AK89] S.B. Akers and B. Krishnamurthy, A group-theoretic model for symmetric interconnection networks. *IEEE Transactions on Computers*, **38**(**4**) pp. 555–566, April 1989.

[AL82] B.W. Arden and H. Lee, A regular network for multiprocessor systems, *IEEE Transactions on Computers*, **31**(**1**) pp. 60–69, January 1982.

[BA84] L. Bhuyan and D.P. Agrawal, Generalized hypercube and hyperbus structure for a computer network, *IEEE Transactions on Computers*, **33**(**3**) pp. 323–333, March 1984.

[DT91] K. Day and A. Tripathy, A comparative study of topological properties of hypercubes and star graphs, Technical Report TR 91–10, Computer Science Department, University of Minnesota, Minneapolis, MN, May 1991.

[Har72] F. Harary, *Graph Theory*, Addison-Wesley, Reading, MA, 1972.

[Kat88] H.P. Katseff, Incomplete hypercube. *IEEE Transactions on Computers*, **C-37** pp. 604–607, 1988.

[Knu73] D.E. Knuth, *The Art of Computer Programming, Volume III.*, Addison-Wesley, 1973.

[LB94] S. Latifi and N. Bagherzadeh, Incomplete star: an incrementally scalable network based on star graph. *IEEE Transactions on Parallel and Distributed Systems*, **5**(**1**), January 1994.

[Lei90] F.T. Leighton, *Introductions to Parrallel Algorithms and Architectures: Arrays, Trees and Hypercubes*, Morgan Kauffman, p. 776, 1990.

[LJD93] S. Lakshmivarahan, J.S. Jwo and S.K. Dhall, Summetry in interconnection networks based on Cayley graphs of permutation groups: a survey, *Parallel Computing*, **19** pp. 361–407, 1993.

[Pra85a] D.K. Pradham, Dynamically restructurable fault tolerant processor network architecture, *IEEE Transactions on Computers*, **34**(**5**) pp. 434–447, May 1985.

[Pra85b] D.K. Pradham, Fault tolerant link and bus network architectures, *IEEE Transactions on Computers*, **34**(**1**) pp. 33-45, January 1985.

[Sch91] I.D. Scherson, Orthogonal graphs for the construction of interconnection networks, *IEEE Transactions on Parallel and Distributed Systems*, **2(1)** pp. 3–19, 1991.

[Sen89] A. Sen, Supercube: an optimally fault tolerant network architecture, *Acta Informatica*, **26** pp. 741–748, 1989.

[SS88] Y. Saad and M.H. Shultz, Topological properties of hypercubes, *IEEE Transactions on Computers*, **37(7)** pp. 867–872, July 1988.

[SS91] S. Sur and P.K. Sriman, Super star: a new optimally fault tolerant network architecture, In "Proceedings of the International Conference on Distributed Computing Systems" (ICDCS-11), pp. 590–597, Texas, 1991.

[SS92a] S. Sur and P.K. Srimani, Incrementally extensible hypercube (IEH) graphs, In "Proceedings of the International Conference on Computers and Communication" (IPCCC-92), pp. 1–7, Phoenix, Arizona, April 1992.

[SS92b] S. Sur and P.K. Srimani, Topological properties of star groups, *Computers & Mathematics with Applications*, **25(12)** pp. 87–98, 1992.