

Topology Embedding on GIN Family

Meenal Borkar¹, Nitin² and Atul Kumar³

^{1,3}School of Engineering and Technology, Ansal University, Gurgaon, India.
Email:meenal.borkar@gmail.com

²Department of CSE and IT, Jaypee Institute of Information Technology, A-10, Sector-62, Noida 201307,
Uttar Pradesh, India.
Email:delnitin@gmail.com

Abstract—This paper discusses about embedding various topologies on GIN family networks. In real multiprocessor / high computing environment, it becomes necessary to check the embedding / mapping of physical topology to logical topology, which is necessarily a graph mapping problem. This paper tries to map / embed frequently used topologies on GIN family network. The promising network variants like 3D-CGIN, CGIN along with original GIN are considered for mapping purpose. GIN variants with size 8 are considered for mapping. The paper presents the comparison of factors related to mapping. The inherent design of GIN makes it well suited for mapping permutation based networks. The mapping penalty is given in terms of additional hop count, to realize connectivity for a path in original network to the base network on which the original network is being mapped. Due to alternate source's use this penalty is found to be negligible.

Index Terms— Interconnection Network Topology, Mapping Topologies, GIN variants, mapping factors, Graph mapping problem.

I. INTRODUCTION

The parallel computing has become a promising technology with the ability to manufacture multicore processors on single chip. These processors communicate with each other through a communication network called Interconnection Network (IN)[1-5][29][34-38]. The INs are generally of two types: Static i.e. Point to Point networks and 2. Dynamic or reconfigurable networks. As the static networks always prove costly for implementation as well as maintenance, dynamic networks are preferred. The processors in dynamic networks are connected to each other with switches. The switches may be made up of either as basic crossbars as or of optical switches. As we can observe here, the processors are connected to each other through switches; these networks are also known as indirect networks. The static networks are known as direct networks.

While designing such huge multiprocessor systems, the communication backbone i.e. IN must be planned carefully. In majority of the implementations dynamic networks are preferred due to their inherent design characteristic. The IN is necessarily a graph made up of processors, switches, memory banks etc. One can easily map or embed another IN, which is again a graph on to it. This mapping / embedding proves not only cost efficient but also make the IN versatile. Multistage Interconnection Networks(MIN), are the dynamic networks where source processors are connected to destination processors / memory banks through multiple intermediate stages of switches. The multiple stages of switches ensure that the MINs to become reconfigurable.

Gamma Interconnection Network (GIN) is a MIN which is based on the concept of redundant paths / alternate paths. This network was proposed by Parker and Raghavendra in 1982[7][8]. Figure 1 shows a typical GIN of size 8. The primary purpose of using redundant paths was to provide improved fault tolerance. Further many researchers analyzed this network for more and more robust working, which led to nearly 20 plus variant networks based on GIN. All these variants[7-17][19-22][30-33] as per their routing method are listed in Table 1. Two routing approaches were used in these networks namely: Distance Tag Routing and Destination Tag Routing[7-17][19-22][24-25][28][30-38][44]. The major focus was always on improvement in fault tolerance. Varied techniques were used by researchers in this direction. Table 2 lists the fault tolerance method used in each of them[7-17][19-22][29-33][44].

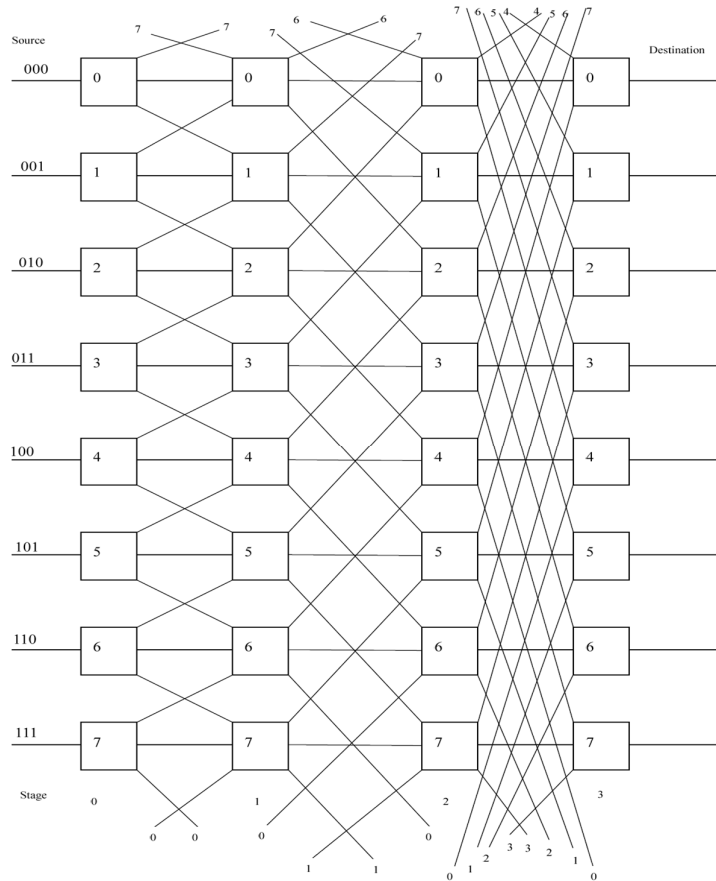


Figure 1. The Gamma Network

The permutations[7][8][26-28] provide the varied mapping of source to destination, which in turn help in implementation of Parallel and Distributed Algorithms. While working on implementation of frequently used permutations on GIN, we observed that mapping other topologies on to GIN can lead to promising results. The algorithms for Distance Tag Routing and Destination Tag Routing can easily be used to map / embed other topologies on to GIN variants. The basic purpose of this study is to analyze GIN's capability in the area of topology mapping.

The paper is organized as follows: Section 1 gives Introduction and the purpose of this research. Section 2 defines the problem of topology mapping as graph mapping problem. Further it provides the details about GIN variants chosen to act as base for mapping problem and the brief idea about the networks to be mapped on them. This section also provides the definitions of the terms used in the work of topology mapping and the definitions of performance metrics used to evaluate the mapping. In Section 3 we present the control algorithms for topology mapping. The algorithms are supported with suitable examples and figures. Section 4 deals with topology mapping parameters and comparison with respect to base network. The section also

TABLE I. GIN VARIANTS ACCORDING TO ROUTING METHOD USED

Sr. No.	Name of Network	Routing Method Used
1.	GIN	Distance Tag Routing
2.	Kappa Network	Destination Tag Routing
3.	Extra Stage GIN	Distance Tag Routing
4.	B-Network	Destination Tag Routing
5.	Balanced GIN	Distance Tag Routing
6.	Mono GIN	Distance Tag Routing
7.	Reliable GIN	Distance tag Routing
8.	Cyclic GIN	Distance Tag Routing , Destination Tag Routing
9.	Partially Chained GIN	Distance Tag Routing
10.	Fully Chained GIN	Distance Tag Routing
11.	3D GIN	Distance Tag
12.	3D-CGIN	Distance Tag Routing, Destination Tag Routing
13.	Incomplete GIN	Twin Tag Routing based on Distance Tag Routing
14.	Incomplete CGIN	Twin Tag Routing based on Distance Tag Routing
15.	4DGIN-1	Distance Tag Routing
16.	4DGIN-2	Distance Tag Routing
17.	NBGIN	Destination Tag Routing
18.	MCDRGN(Additional link at initial stage)	Distance Tag Routing
19.	MCDRGN (Additional link at initial and intermediate stages)	Distance Tag Routing
20.	DRGIN	Distance Tag Routing, Destination Tag Routing
21.	GIN with Alternate Source	Distance Tag Routing, Destination Tag Routing

TABLE II. GIN VARIANTS ACCORDING TO FAULT TOLERANCE METHOD USED

Sr. No.	Fault Tolerance Method Used	Name of Network/(s)
1.	Adding Extra Stage	Extra Stage Gamma Network
2.	Providing Back Links	B-Network
3.	Providing Extra Link	Balance Gamma Network, PCGIN, FCGIN, MCDRGN
4.	Changing Interconnection Pattern	Reliable Gamma Network, Monogamma Network, Cyclic Gamma Interconnection Network, Balanced Modified Gamma Network
5.	Combining the Switching Elements	3DGIN, 4DGIN-1, 4DGIN-2
6.	Providing Alternate Source	3D-CGIN, GIN with Alternate Source

presents the concept of mapping penalty in terms of additional Hop Count. The *experimental* setup is also covered in this section. Section 5 presents conclusions. The references are given at the end.

II. BACKGROUND

As it was mentioned in Section 1, there are 20 plus GIN variants available. Among these we have chosen GIN, CGIN and 3D-CGIN as they found more suitable and proved beneficial in terms of mapping penalty. In this section we would present the terms and terminology used, followed by brief information about CGIN and 3D-CGIN[22][24-25]. The selected topologies for mapping purpose are also discussed in brief in this

section.

A. Terms and Terminology

Definition 1: Physical Topology

This is the topology / arrangement of the components of the multiprocessor system on which another topology is to be matched. The Physical Topology is defined as $G=(V,E)$, where V is the set of components of the network and E is the set of edges showing connections between the elements of V .

Definition 2: Logical Topology

This is the topology / arrangement of the components of the network which is to be matched on Physical Topology. The Logical Topology is defined as $G'=(V',E')$, where V' is the set of components of the network and E' is the set of edges showing connections between the elements of V' .

Definition 3: Topology Mapping

Topology Mapping is the process of realizing logical topology onto physical topology. This mapping is achieved by simulating G' over G . Generally number of components in V and V' are same, but the sets E and E' do differ. After mapping, we get a graph $G''=(V'', E'')$ denoting the mapped topology onto physical topology.

B. Cyclic Gamma Interconnection Network

In n-stage Gamma Network, the connection pattern can be represented as $2^0, 2^1, 2^2, \dots, 2^{n-1}$. In CGIN [8] the connection patterns were altered by repeating any stage at initial and end connections. This alteration provided multiple disjoint paths, guaranteeing one arbitrary fault tolerance. It also guaranteed multiple paths between any pair of source and destination. The resulting network depicted cyclic nature.

A $CGIN_n^\gamma$ is a CGIN with n stages and the connecting patterns between first two stages and last two stages being 2^γ where $0 \leq \gamma \leq (n-2)$. So, the connecting patterns between stages can be ordered as

$$2^\gamma, 2^{\gamma+1}, 2^{\gamma+2}, \dots, 2^{n-3}, 2^{n-2}, 2^0, 2^1, 2^2, \dots, 2^{\gamma-1}, 2^\gamma$$

The stages of $CGIN_n^\gamma$ are numbered 0 to n and the connecting patterns are based on plus-minus $2^{(\gamma+i)} \pmod{(n-1)}$ functions. Figure 2 shows a typical $CGIN_3^0$.

Each request in $CGIN_n^\gamma$, carries a routing tag of n digits. The weight of each digit is determined by the connecting pattern. For a tag digit d_i , the weight is determined by the formula: $\pm 2^{(\gamma+i)} \pmod{(n-1)}$ if d_i is ± 1 . The routing complexity of CGIN is same as that of GIN. CGIN reduces the pin count, as it uses 2^γ instead of 2^{n-1} connecting pattern. It reduces the total layout area as well thus, achieving reduction in cost.

CGIN uses the destination tag routing and re-routing to tolerate any arbitrary single fault. It does not provide strong re-routability. Strong re-routability implies that the packet can be re-routed at every stage. CGIN provides at least 2 disjoint paths between any pair of source and destination. However, packet delivery can fail, if the source is faulty.

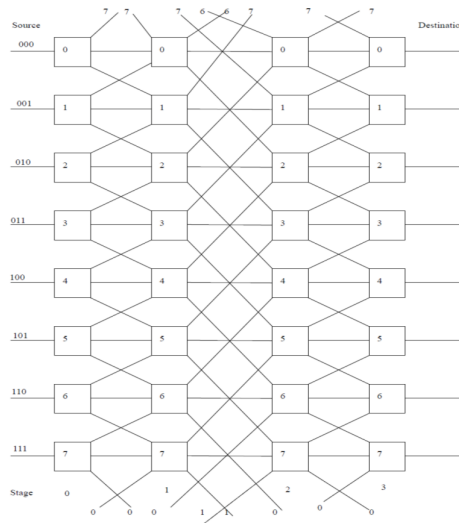


Figure 2 $CGIN_3^0$

C. 3D-CGIN: A 3 disjoint Cyclic Gamma Interconnection Network with Alternate Source

In attempt to provide more than 2 disjoint paths, authors proposed a modified CGIN with alternate source at the initial stage. This network provided at least 3 disjoint paths between every communicating pair, hence named as 3D-CGIN. The network topology is similar to CGIN except the elements at initial stage use 2 x 3 switches. The added link at source provides connection to the alternate source located at $N/2$ distance, where N is the network size. This link is dual direction link. The Figure 3 shows a typical 3D-CGIN with size 8. The network uses distance tag as well as Destination Tag routing. It also proposed the pre-computing of routing tag methods, which ensure advanced computation of fault free routing tags.

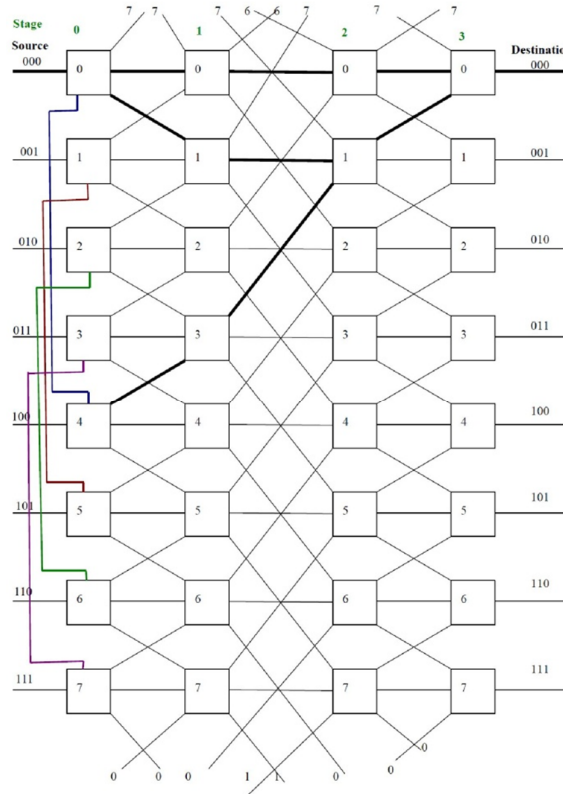


Figure 3. 3D-CGIN with Alternate Source, size $N = 8$

D. Selected Topologies for Mapping

We have chosen popular topologies, which are being used in multiprocessor systems. We selected Complete Binary Tree, Cube, de Bruijn Network, and Butterfly network for mapping purpose. These networks will be mapped on different variants of GIN as per their characteristics. In this subsection, we provide brief information about these networks[39-43][45-51].

Complete Binary Tree: Complete Binary Tree is a Tree topology interconnection network. It is a rooted tree arrangement. Generally the rooted tree arrangements are popularly used in parallel and distributed processing, as they work with divide-and-conquer technique. The CBT topology is popularly used to implement the parallel algorithms based on divide-and-conquer. Figure 4 below shows a CBT of size $N=8$. [43].

Three Dimensional Cube Topology: In an n -cube network, the network size is $N=2^n$. [42] The network is also known as hypercube. In our study we used $n = 3$ to indicate 3 dimensional cube with size 8. In this network each node is connected with 3 another nodes. In an n -cube, individual nodes are uniquely identified by n -bit addresses ranging from 0 to $N-1$. Given a node with binary address d , this node is connected to all nodes whose binary addresses differ from d in exactly 1 bit. For example, in a 3-cube, in which there are eight nodes, node 7 (111) is connected to nodes 6 (110), 5 (101), and 3 (011). Figure 5 shows a typical 3D-Cube network.

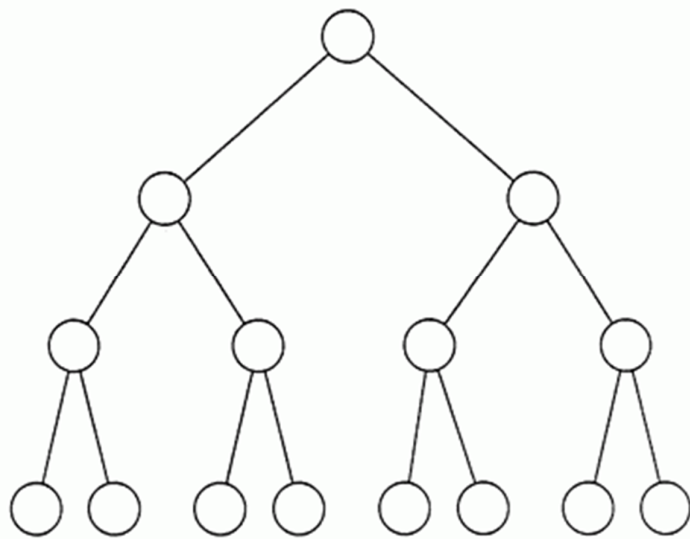


Figure 4. Complete Binary Tree of size N = 8

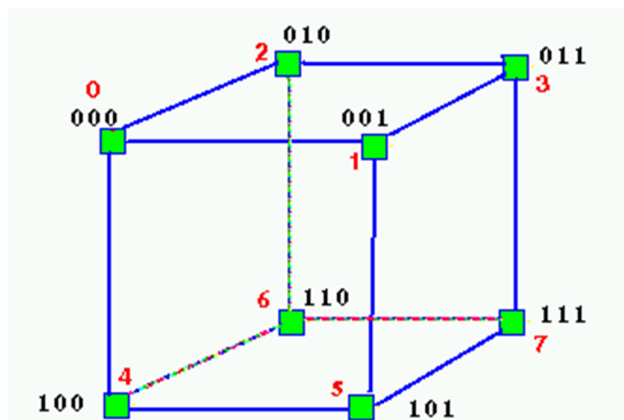


Figure 5. Three Dimensional Cube Topology of size N = 8

de Bruijn Network: Generally the de Bruijn Network is defined as a graph[41]. A de Bruijn graph of order n , has 2^n nodes labeled with a bit representation of length n of the numbers 0 to $2^n - 1$. The vertices are connected if and only if the label of one is left-shifted or right-shifted label of the other. In simple words, it is the left- or right-shifted label of the other and differs correspondingly in the first or last bit. This property generates 4 edges at each node and a graph with 2^{n+1} edges. In our study we have considered the de Bruijn network of order 3, which has 8 nodes. Figure 6 shows a typical de Bruijn Network of size 8.

Butterfly Network: The d -dimensional butterfly $BF(d)$ is a graph with node set $V = (d+1) \cdot 2^d$ and an edge set $E = E1 \cup E2$ where $E1 = \{(i, \alpha), (i+1, \alpha)\} | i \in [d], \alpha \in [2^d]\}$ and $E2 = \{(i, \alpha), (i+1, \beta)\} | i \in [d], \alpha, \beta \in [2^d], \alpha$ and β differ only at i^{th} position}. In our study we have selected Butterfly Network of size 8. Figure 7 shows a typical Butterfly Network. [40]

E. Performance Metrics used for evaluation of topology embedding

In order to evaluate the topology embedding / mapping, three metrics are commonly used. In this subsection we define and give brief description about them. [39]

1. Dilation: It is defined as the maximum number of link / edges in the embedding that is mapped in one link in the original network. Here the embedding refers to the logical topology and original network refers to physical topology.
2. Congestion: It is defined as the maximum number of edges mapped into one edge of embedding.

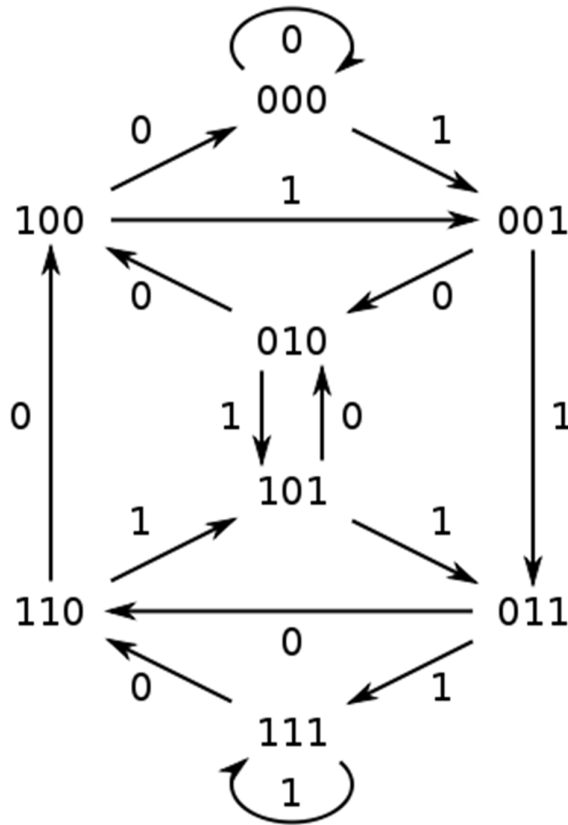


Figure 6. de Bruijn Network Topology of size N = 8

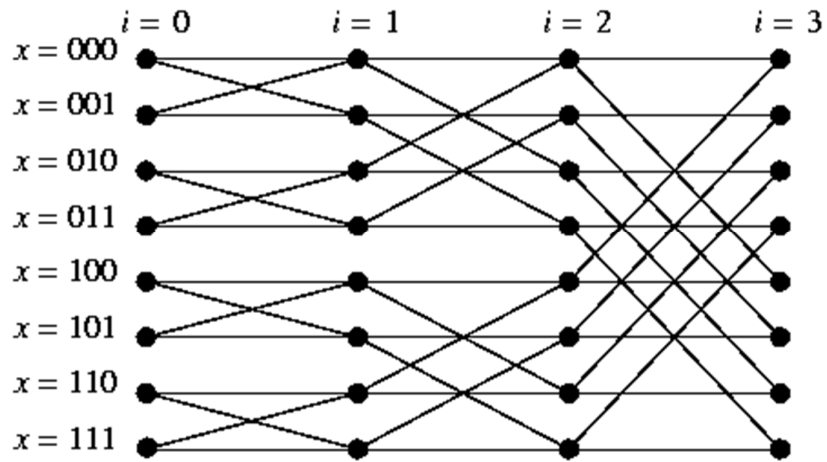


Figure 7. Butterfly Network Topology of size N = 8

3. Scaling: It is the ratio of the number of nodes in the embedding to the number of nodes in the original network. Some researchers had referred scaling as expansion.

III. PROPOSED WORK

In this section we present the algorithms used to map the logical topology on to GIN variants of size N=8. As

GIN variants are based on alternate paths method for routing, the selection of routing tag is non-deterministic process. Therefore we have assumed that the best generated path is used to map a source to destination. The subsections discuss the mapping of logical topology to specific physical topology with the help of algorithms and figures.

A. Mapping Ordinary Butterfly

As observed from the topology description of ordinary Butterfly in section 2, the connection patterns between the stages are in the range from distance 1 to distance 4. This observation leads us to a conclusion that GIN can be used to map ordinary Butterfly, as its connection patterns are exactly in the same sequence. The only change needed in the physical topology i.e. GIN, is disabling certain links.

For connections between stage 0 and 1, the uplinks and downlinks are disabled for every alternate node respectively. For connections between stage 1 and 2, the uplinks and downlinks are disabled for every 2 alternate nodes respectively. For connections between stage 2 and 3, the uplinks and downlinks are disabled for every 4 alternate nodes respectively. The Butterfly network is obtained. The algorithm presented below does the same thing. We use circuit switching type of routing during mapping, which suggests that all the links must be reserved before transmitting the packets. The GIN will then behave exactly like Ordinary Butterfly network.

Algorithm 1: Mapping Ordinary Butterfly on to GIN

Input: *The topology of GIN and Ordinary Butterfly of size $N = 8$*

Output: *Mapped Ordinary Butterfly onto GIN*

Method:

1. *Read the topology of GIN, where $PE[up]$, $PE[str]$ and $PE[dn]$ denote the uplink, straight and down connection respectively.*
2. *Let $cnbtstg=1$ [$cnbtstg$ denote connections between stages. Currently it indicates connection between stage 0 and 1]*
3. *for stage 0 to 3*
 - a. *if $cnbtstg == 1$ then*
 - i. *for PE 0 to 7*
 1. *if PE is even then set $PE[up] = 999$, where 999 indicates the link is disabled*
else set $PE[dn] = 999$, where 999 indicates the link is disabled
[End of if]
- b. *if $cnbtstg == 2$ then*
 - i. *for PE 0 to 7*
 1. *if PE is 0, 1, 4 or 5 then set $PE[up] = 999$, where 999 indicates the link is disabled*
else set $PE[dn] = 999$, where 999 indicates the link is disabled
[End of if]
- c. *if $cnbtstg == 3$ then*
 - i. *for PE 0 to 7*
 1. *if PE is 0, 1, 2 or 3 then set $PE[up] = 999$, where 999 indicates the link is disabled*
else set $PE[dn] = 999$, where 999 indicates the link is disabled
[End of if]

The Algorithm 1 ensures that, all the respective nodes have up and down links disabled respectively. Figure 8

shows a GIN mapping Ordinary Butterfly. We have used the Connected-to-Set method for this purpose. Interested readers can find more details about Connected-to-Set pre-computing tags method from a paper titled “Network Status Aware Routing in 3D-CGIN” authored by Meenal Borkar and Nitin.

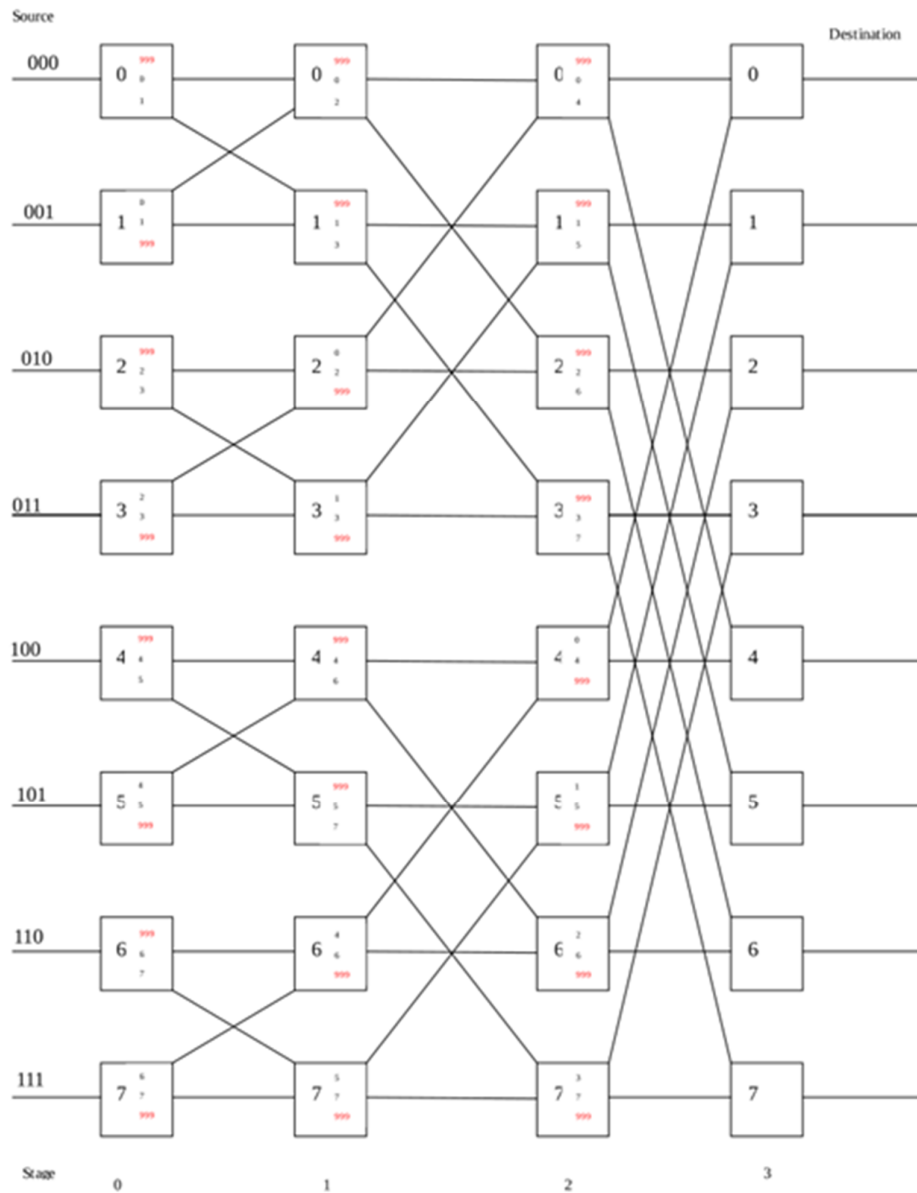


Figure 8. Butterfly Network mapped on GIN size $N = 8$ with the help of CtS

B. Mapping 3D Cube

After observing the 3DCube topology, we can see that each node is connected with 3 other nodes directly with single link. The node can reach to other nodes which are reachable using 2 or 3 links connected to each other. In order to map Cube, we found the networks with cyclic connection patterns useful. During mapping the Cube on CGIN, we found that for some of the destinations for two nodes, all the paths were exhausted. Therefore we decided to use 3D-CGIN where the alternate path can be used to route packets for those two communicating pairs. The circular interconnection patterns in 3D-CGIN allowed the routing of packets to nodes which are reachable through 2 more consecutive links. In this implementation the path reservation is

not needed, rather packet switching technique can be used. Both the routing techniques, namely – Distance Tag Routing and Destination Tag Routing can be used here to map Cube on 3D-CGIN.

C. Mapping Complete Binary Tree

The Complete Binary Tree topology is very popular in parallel architectures which are specifically designed to work with divide and conquer algorithms. We have used GIN with reverse connections for mapping this topology. The CtS is used to enable and disable paths well in advance before sending the packets. Figure 9 below shows the mapping of Complete Binary Tree on GIN with reverse connections.

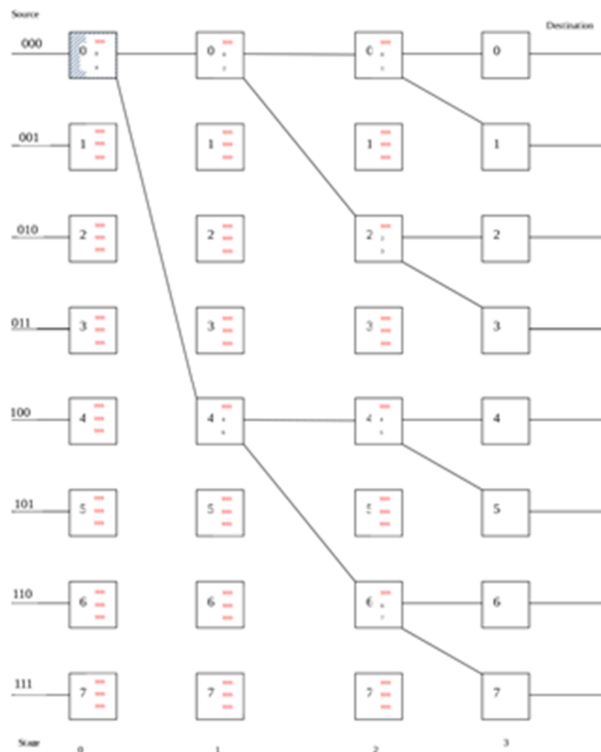


Figure 9. Complete Binary Tree mapped on GIN with reverse connections size $N = 8$ with the help of CtS. Hatched node is the root node for this CBT

It is evident from the above figure, that the connection capability of GIN with reverse connections is underutilized. We can map 4 CBTs on to GIN with reverse connection pattern with introduction of time delay in between. Figure 10 below shows this scheme.

D. Mapping de Bruijn Network

In de Bruijn Network, every node is connected with two nodes through direct links. Every node is connected with all other nodes through indirect links. Again the cyclic connection network patterns will help us to map this network in multiple passes. Due to this reason we have selected CGIN as our base network. The Distance Tag Routing and Destination Tag Routing can be used to reach the destinations.

IV. RESULTS AND COMPARISON

In this section, we compare the performance metrics of mapping these networks on GIN variants. We have used dilation, congestion and scalability as our performance metrics. The experimental setup and assumptions are also presented in this section.

A. Experimental Setup and Assumptions

The simulation for topology mapping is written in JavaScript. We assumed fault-free environment for the simulation. The distance tag routing is primarily used to generate the routing tags. In GIN the selection of

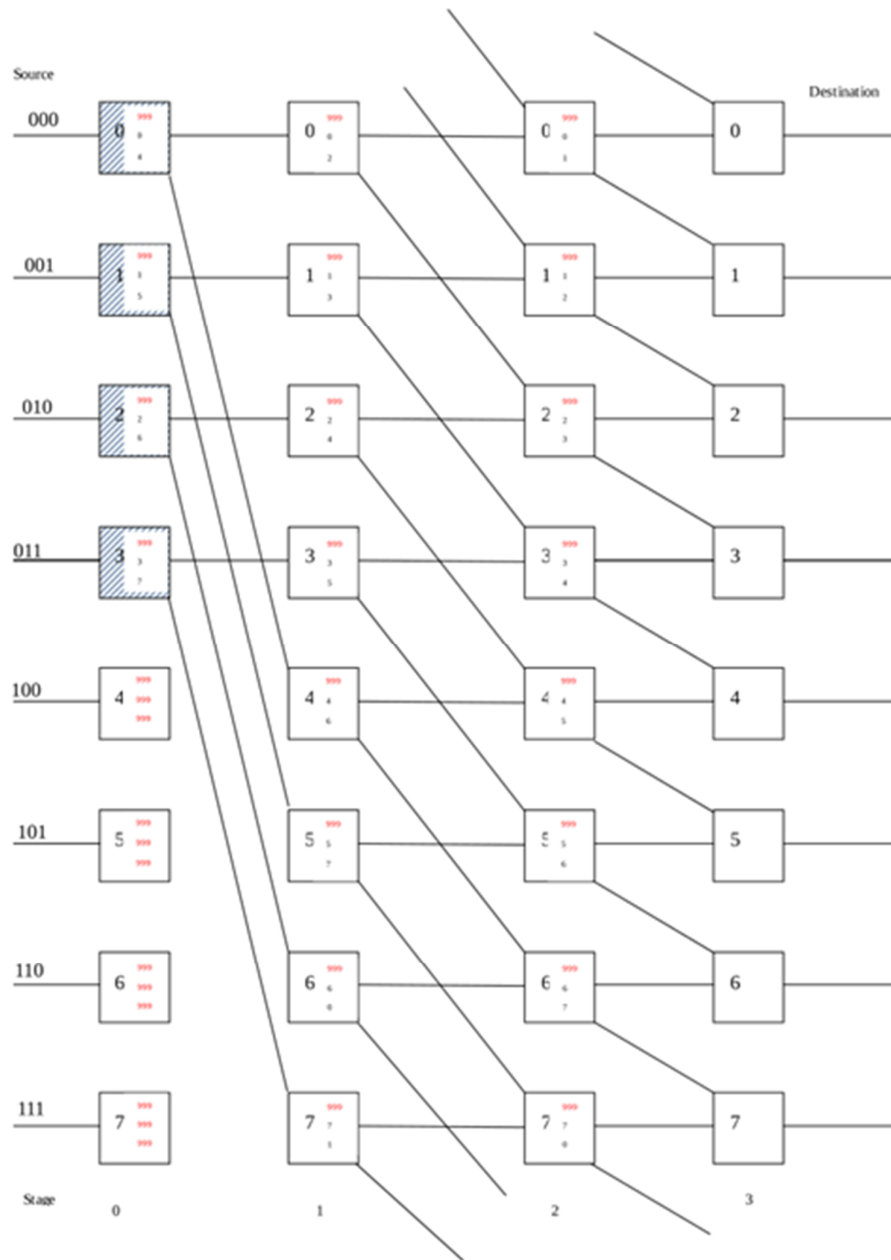


Figure 10. Complete Binary Tree mapped on GIN with reverse connections size $N = 8$ with the help of CtS. Hatched nodes are the root nodes for these CBTs

routing tag is a nondeterministic choice, so we have assumed that the best suited tag is selected every time. The alternate source is used whenever all the links from a particular node are exhausted. The simulations are tested on a Dell Vostro Quadcore system running on 64 bit Ubuntu Linux.

B. Comparison for evaluation of mapping using performance metrics

As mentioned in section 2, we have evaluated the mapping using three performance metrics. They are dilation, congestion and scaling. The dilation means maximum number of links in embedding / logical network that is mapped in one link in the original network. By congestion we mean maximum number of edges mapped into one edge in the embedding. Expansion is the ratio of number of nodes in embedding to

that in original network. In our study for majority of networks we got constant performance metrics values. Table 3 below shows the observed values for the same.

TABLE III. PERFORMANCE EVALUATION OF MAPPING

Sr. No.	Metrics	Butterfly	3D-Cube	CBT	De Bruijn
1	Dilation	1	4	2	$\log_2 N$
2	Congestion	1	2	1	2
3	Scalability	1	4	$N/(N-1)$	4

V. CONCLUSION

In this paper we have presented the work for mapping or embedding some frequently used networks on GIN variants. We have selected Ordinary Butterfly Network, 3D-Cube, de Bruijn Network and Complete Binary Tree. All these networks are popularly used in parallel computing environment. During our study we found that, the methods of pre-computing tags play crucial role, in fixing the routes for topology mapping. The study also found that in many cases the Distance Tag Routing and Destination Tag Routing can readily applied to map a network on to GIN variant. The alternate source and cyclic connection patterns proved beneficial as usual, and allowed easier mapping. The evaluation of all these mappings is done using three metrics. The paper presented the detailed implementation of mapping with algorithms and figures.

REFERENCES

- [1] Advanced Computer Architecture, Kai Hwang, Tata-McGraw Hill Publications.
- [2] Interconnection Networks, William J. Dally, Morgan-Kaufmann Publication
- [3] An Introduction To Parallel Computing, Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, Addison Wesley.
- [4] Parallel Programming, Barry Wilkinson, Pearson Eductaion.
- [5] T. Y. Feng, "A Survey of Interconnection Networks", IEEE Transactions on Computers, December 1981
- [6] G. B. Adams III, D. P. Agrawal, and H. J. Siegel, "A Survey and Comparison of Fault-Tolerant Multistage Interconnection Networks", IEEE Transactions on Computers, June 1987
- [7] D. S. Parker, and C. S. Raghavendra, "The Gamma Network: A Multiprocessor Interconnection Network With Redundant Paths", IEEE Transactions on Computers, Los Alamitos 1982
- [8] D. S. Parker, and C. S. Raghavendra, "The Gamma Network", IEEE Transactions on Computers, Vol. c-33, No. 4, April 1984
- [9] S. C. Kothari, G. M. Prabhu and Robert Roberts, "The Kappa Network with Fault-Tolerant Destination Tag Algorithm", IEEE Transactions On Computers, Vol.37 No 5, 1988
- [10] K.Y. Lee, and W. Hegazy, "The Extra Stage Gamma Network", IEEE Transactions on Computer, Vol. 37, No. 11, November 1988
- [11] K. Y. Lee, and H. Yoon, "The B-Network: A Multistage Interconnection Network With Backward Links", IEEE Transactions on Computer, Vol. 39, No. 7, July 1990
- [12] R. Venkatesan, and H. T. Mouftah, "Balanced Gamma Network-A New Candidate For Broadband Packet Switch Architectures", IEEE Transactions on Computer, INFOCOM 1992
- [13] C. W. Chen, N. P. Lu, T. F. Chen, and C. P. Chung, "Fault Tolerant Gamma Interconnection Networks By Chaining", IEE Proceedings - Comput. Digit. Tech, Vol. 147, No. 2, March 2000
- [14] P. J. Chuang, "Creating a Highly Reliable Modified Gamma Interconnection Network Using a Balance Approach", IEE Proceedings - Comput. Digit. Tech, Vol. 145, No. 1, January 1998
- [15] N. F. Tzeng, P. J. Chuang, and C. H. Wu, "Creating Disjoint Paths In Gamma Interconnection Networks", IEEE Transactions on Computer, Vol. 42, No. 10, October 1993
- [16] P. J. Chuang, "CGIN: A Modified Gamma Interconnection Network with Multiple Disjoint Paths", IEEE Transactions on Computer, 1994
- [17] C. W. Chen, N. P. Lu, and C. P. Chung, "3-Disjoint Gamma Interconnection Network", The Journal of Systems and Software, 2003
- [18] Darwin Rau, Jose A. B. Fortes, and Harward Jay Siegel, "Destination Tag Routing Techniques Based on a Stage Model for the IADM Network", IEEE transactions on computers, VOL 42, no. 3, 1992
- [19] C.W. Chen, C. J. Ku, and C. H. Chang, "Design Schemes and Performance Analysis of Dynamic Rerouting Interconnection Networks for Tolerating Faults and Preventing Collisions", Parallel and Distributed Processing and Applications, ISPA, 2005
- [20] Zhen Chen, "A Class of Incomplete Gamma Interconnection Network", Available at: www.researchgate.com
- [21] Meenal A. Borkar, "A Survey of Fault Tolerance Techniques Used in GIN", in National Conference EEC, 2010
- [22] Meenal A. Borkar, and Nitin, "3D-CGIN: A 3 Disjoint Paths CGIN with Alternate Source", in ACC, 2011

- [23] Barlik, P. K., "FIR Filter IC Design Using Redundant Binary Number Systems", M.Tech Thesis, NIT Rourkela, India, 2011
- [24] Meenal A. Borkar, "3D-CGIN: A 3Disjoint Paths CGIN with Alternate Source", M.Tech Dissertation, UTU Dehradun, India, 2011
- [25] Meenal A. Borkar, and Nitin, "Network Status Aware Routing in 3D-CGIN", in ICCCS, 2012
- [26] A. Varma, and C. S. Raghavendra, "On Permutations Passable by the Gamma Network", Journal of Parallel and Distributed Computing, Vol 3, 1986
- [27] Sandeep Sharma, "On Permutation Capabilities of Fault Tolerant Multistage Interconnection Networks", IJCSI, Vol. 9, Issue 6, No 3, 2012
- [28] Meenal A. Borkar and Bhupendra Bhadana, "On Performance Evaluation Parameters of Multistage Interconnection Networks", in Second ICSET, Ansal University, Gurgaon, India, 2014
- [29] A. Avizienis, "Signed-Digit Number Representations for Fast Parallel Arithmetic", IRE Transactions, EC-10, 1961
- [30] Interconnection Networks: An Engineering Approach, J. Duato, Morgan-Kaufmann Publication
- [31] C. Q. Chaoyang, "A minimal cost dynamic rerouting gamma network", Available At: <http://wr.cyut.edu.tw>
- [32] S. Rajkumar and Neeraj K. Goyal, "Design of 4-disjoint gamma interconnection layouts and reliability analysis of gamma interconnection networks", Journal of Supercomputing, 2014
- [33] C. W. Chen, C. P. Chung, "Fault Tolerant Gamma Interconnection Networks without backtracking", Journal of Systems and Software, 2001
- [34] Nitin, Rajan Vaish and Utkarsh Srivastava, On a Deadlock and Performance Analysis of ALBR and DAR Algorithm on X-Torus Topology by Optimal Utilization of Cross Links and Minimal Lookups, Journal of Supercomputing, Springer, Volume 59, Number 3, December 2010, pp. 1252-1288.
- [35] Nitin and Durg Singh Chauhan, Stochastic Communication for Application Specific Networks-on-Chip, Journal of Supercomputing, Springer, Volume 59, Number 2, September 2010, pp. 779-810.
- [36] Nitin and Durg Singh Chauhan, Comparative Analysis of Traffic Patterns on k-ary n-tree using Adaptive Algorithms based on Burton Normal Form, Journal of Supercomputing, Springer, Volume 59, Number 2, June 2010, pp. 569-588.
- [37] Nitin Rakesh and Nitin, Analysis of Multi-Sort Algorithm on Multi-Mesh of Trees (MMT) Architecture, Journal of Supercomputing, Springer, Volume 57, Number 3, March 2010, pp. 276-313.
- [38] Nitin, Shruti Garhwal and Neha Srivastava, Designing a Fault-tolerant Fully-chained Combining Switches Multi-stage Interconnection Network with Disjoint Paths, The Journal of Supercomputing, Springer, Volume 55, Number 3, October 2009, pp. 400-431.
- [39] Q. Mohammad, A. Alamouh, S. Basem, M.M.A. Assaf et al, "Embedding Bus and Ring in to Hex-Cell Interconnection Network", Vol 7, No. 3, IJCNC, 2015
- [40] Roger Wattenhofer, "Principles of Distributed Computing", Available at <http://82.130.102.95/lectures/ss05/distcomp/lecture/chapter5.pdf>
- [41] D. L. Quesnel, Course Notes, "de Bruijn Network", 1995
- [42] Text Book for Interconnection Networks, Available at <http://www2.cs.siu.edu/~cs401/Textbook/ch5.pdf>
- [43] Junming Xu, Topological Structure and Analysis of Interconnection Networks, Springer Science and Business Media, 2013
- [44] Meenal Borkar, Nitin, "Realizing frequently used permutations on gamma interconnection network's family networks with the help of alternate source", Vol 71, Number 12, Journal of Supercomputing, 2015
- [45] M. Hamdi, "Embedding hierarchical networks into the Hypercube", Proceedings of the 37th Midwest symposium on Circuits and Systems, 1994
- [46] Dajin Wang, "On embedding Hamiltonian Cycles in Crossed Cubes", IEEE Transactions on Parallel and Distributed Systems, Vol 19, Issue 3, 2008
- [47] Jung-Hwan Chang; Jinsoo Kim, "Ring embedding in faulty (n,k)-star graphs", Proceedings of Eighth International conference on Parallel and Distributed Systems, 2001
- [48] N. Gopalakrishna Kini, M. Sathish Kumar, H. S. Mruthyunjaya, "A Torus Embedded Hypercube Scalable Interconnection Network for Parallel Architecture", IEEE International Conference on Advanced Computing, 2009
- [49] Haun-Chao, Jen-Chih Lin, "Embedding a complete binary tree into a faulty supercube", 3rd International conference on Algorithms and Architectures for Parallel Processing, 1997
- [50] Xiaojun Shen, Weifa Liang, Qing Hu, "On embedding between 2D meshes of the same size", IEEE Transactions on Computers, Volume 46, Issue 8, 1997
- [51] K. Gupta, W. J. Dally, Topology Optimization of Interconnection Networks, IEEE Computer Architecture Letters, Vol. 5, Issue 1, 2006