Research Journal of Applied Sciences, Engineering and Technology 11(2): 198-214, 2015
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: April 28, 2015 Accepted: May 22, 2015 Published: September 15, 2015

Fault Tolerance in Interconnection Network-a Survey

[1]Laxminath Tripathy, [2]Devashree Tripathy and [3]C.R. Tripathy
[1]Department of CSE, Orissa Engineering College Bhubaneswar, Odisha, India
[2]Central Electronics Engineering Research Institute (CEERI) Pilani, India
[3]Sambalpur University, Sambalpur, Odisha, India

Abstract: Interconnection networks are used to provide communication between processors and memory modules in a parallel computing environment. In the past years, various interconnection networks have been proposed by many researchers. An interconnection network may suffer from mainly two types of faults: link faults and/or switch fault. Many fault tolerant techniques have also been proposed in the literature. This study makes an extensive survey of various methods of fault tolerance for interconnection networks those are used in large scale parallel processing.

Keywords: DFA property, dynamic fault-tolerance, fault model, interconnection network, MIN, static fault tolerance

INTRODUCTION

Interconnection Network (ICN) is used to interconnect processor to processor and processor to memory in a network. Interconnection network plays a crucial role in enhancing the performance of a parallel system in which multiple processor have direct access to shared memory.

In the past many researchers have proposed various types interconnection networks and most of the networks are discussed in (Feng, 1981; Adams III *et al*., 1987; Skillicorn, 1988 and Tripathy and Adhikari, 2011; Skillicorn, 1988; Street and Wallis, 1977; Leiserson, 1985; Kamiura *et al*., 2000, 2002) and more network discussed subsequently. Based upon the technique of interconnection, an interconnection network may be designated either as dynamic or static. Static networks consist of point-to-point communication links among processing nodes and are also referred to as direct networks. Dynamic networks are built using switches and communication links. Dynamic networks are also referred to as indirect networks.

Most of the dynamic interconnection networks comprise of switches and links between the input and output terminals. The signal enters the network through the input port and leaves from the output port. A network with input port A and output port B is represented as A×B network. A dynamic interconnection network may contain either a single stage or multiple stages through which data/signal pass from the source to the destination. However, a static interconnection on the other hand, consists of an interconnection of stand-alone processors. Among those interconnection networks, some are designed to tolerate faults and others do not.

However, fault tolerance capability of an interconnection network enhances the overall reliability of the parallel system and adds to its performance improvement (Dash *et al*., 2012).

The faults associated with a parallel system can be of many types and accordingly, the techniques to embed fault tolerance into an interconnection network can be different. The fault tolerant capability of any interconnection network ensures that the network is able to provide service in presence of faulty components.

Our discussions here also include how various interconnection networks tolerate a single fault or multiple faults either by adding extra hardware or rerouting the packets. Apart from various regular multi stage interconnection networks proposed for parallel systems other networks like fat tree (Leiserson, 1985), Siamese-twin fat tree (Sem-Jacobsen *et al*., 2005), Modified Fault tolerant Double Tree (MFDOT) (Sengupta and Bansal, 1998) hyper cube (Leighton, 1992) have been included and discussed how these networks tolerate faults.

This study first makes an in-depth study of various types of faults that may affect the performance of an interconnection network. Next, we discuss the various fault tolerance techniques those can be embedded in the networks so as to make them fault free.

This survey portrays the diversity of fault tolerant MINs and other networks in terms of fault tolerance. The relative merits of the fault tolerant interconnection network are studied.

FAULT TOLERANCE TECHNIQUES

The fault may be either at switch level (i.e., switch fault) or at link level (i.e., link fault). A fault can be

Corresponding Author: Laxminath Tripathy, Department of CSE, Orissa Engineering College Bhubaneswar, Odisha, India

either permanent or transient. Otherwise the fault is assumed to be permanent. The fault tolerance is defined with respect to a fault tolerant model which can have two parts. The fault model characterizes all faults that are assumed to occur in the network. The fault tolerance criterion requires that sufficient conditions should met so that the network tolerates faults. The Dynamic Full Access (DFA) property of a network states that each of its inputs can be connected to any one of its outputs in a finite number of passes through the network. This serves as the important criterion for fault tolerance. So this property is studied in presence of faults.

Fault tolerance can be either static fault tolerance or dynamic fault tolerance. It can be achieved at various levels in a complex system. In static fault tolerance, during routing of message/signal if any link or switch lying in the routing path gets failed the tolerance can be achieved by reconfiguring or restarting network and rerouting the packet in a new path. In dynamic fault tolerance, faults can be tolerated dynamically without restarting the network which have discussed in (Sem-Jacobsen *et al*., 2005, 2011; Kim *et al*., 1997; Theiss and Lysne, 2006; Sem-Jacobsen *et al*., 2006).

We assume fault diagnosis to be available as needed with respect to the surveyed ICNs and do not discuss it further. The techniques for fault-tolerant design can be categorized by whether they involve modification of the topology (graph) of the system. The three well-known methods that do not modify topology are error-correcting codes, bit-slice implementation with spare bit slices and duplicating an entire network (this changes the topology of the larger system using the network). These approaches to fault tolerance can be applied to ICNs. Over the years number of techniques have also been developed to suit to the nature of ICNs and their use. Our survey here explores these methods in particular in a systematic order.

The networks that are surveyed here are ordered roughly by the hardware modifications made to provide redundancy, from less to more extensive. Many possible techniques do exist for fault tolerance. Those include adding an extra stage of switches, varying switch size, adding extra links and adding extra ports. The technique of chaining switches within a stage so that data can sidestep a faulty switch is discussed in detail in this study. Some of the techniques are also based upon new ICN by adding extra hardware.

Fault-tolerance in single-stage inter connection networks: A single stage beta interconnection network is proposed by Huang and Chen (1987) and shown in Fig. 1 where the single stage switches are used for connecting the processing elements. Such a network is fault tolerant by connection of extra switches at input and output part.

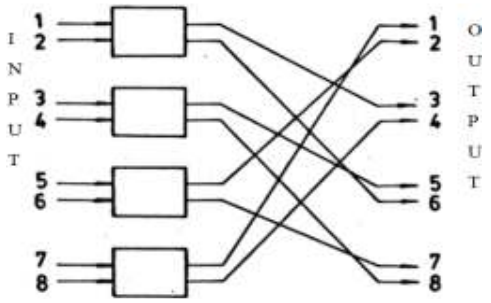


Fig. 1: Single stage Beta interconnection network

An n×n single-stage Beta network is composed of n/2 number of 2×2 Switching Elements (SE). The single stage Beta network with four switching elements as shown in Fig. 1 can be imparted fault tolerance.

In such a network has two states, referred to as through and cross state, corresponding to the two possible permutations of its input terminals. There is a control line associated with each input terminal to control which output the input terminal is to be connected. Data are routed to their destinations by recirculating through the network. The faults can be tolerated by allowing data to recirculate in the network through several more passes. Two parameters have taken into account to evaluate the network i.e., communication delay (d) and degree of fault tolerance (k). It has been shown in beta interconnection network that $k+1 \leq d$. The condition for optimal fault tolerance is $k = d-1$. The criterion for fault tolerance in Beta networks is called the Dynamic Full Access (DFA) property (Shen and Hayes, 1984). The fault tolerance of a Beta network is defined as its ability to maintain DFA properties in spite of the presence of stuck-at faults in its SE's. A Beta network can be made more faults tolerant if it is able to tolerate a large number of faulty SE's. A Beta network with DFA property is k-fault tolerant if the failure, either stuck-at-through or stuck-at-cross, or any k or fewer SE's do not destroy the DFA property, where k is called the Fault Tolerant (FT) parameter of the Beta network.

A fault tolerant scheme for multistage interconnection network: Multistage Interconnection Networks (MINs) are a class of high-speed computer networks usually composed of Processing Elements (PEs) on one end of the network and Memory Elements (MEs) on the other end, connected by switching elements (SEs). The switching elements themselves are usually connected to each other in stages, hence the name a Multistage Interconnection Network (MIN) called Baseline interconnection network of 8 input and 8 output (i.e., 8×8) is shown in Fig. 2. The detail techniques for tolerating faults are discussed by Tzeng *et al*. (1985). The techniques are applicable to these types of MINs which have unique path between every source and destination pair. A Baseline Interconnection

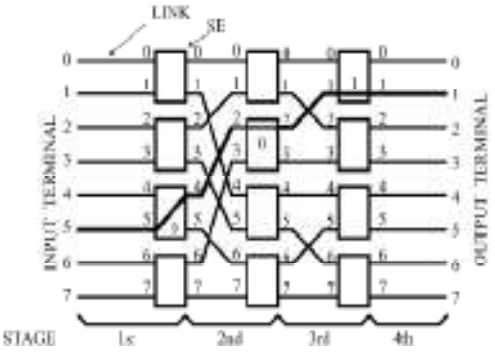


Fig. 2: Baseline interconnection network

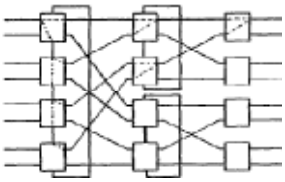


Fig. 3: Illustration of fault-tolerance in MINs by adding extra links

Network (BICN) is taken as example. In a Baseline interconnection network there is only one link between every source and destination pair. So if fault appear in path then communication would not be possible.

Fault-tolerance in MINs by adding extra links: Tzeng *et al*. (1985) proposed a technique of creating multiple paths between each input/output pair through extra links between the switching elements in the same stage. As a result if any fault arises in any link between source destinations then an alternative path will be chosen. The addition of extra link in Base line ICN is illustrated in Fig. 3. Here the switching elements are chaining together to form multiple path which is used to provide fault tolerance capability to the network.

In order to provide fault tolerance to the switches at input and output stage of Base line ICN each SE at the last and first stage is made a complete chain as shown in Fig. 4. According to this scheme the last stage of the network can tolerate two faulty outputs in each switching element without losing the connectivity. Hence it can tolerate at most N faults in the last stage. At the input stage each system component has to access two input elements. So the said network tolerates at most when half of input elements are being faulty. Overly the number of faulty elements the entire network can tolerate is $N\log_2 N+1$ where N is the number of inputs/outputs.

However it cannot tolerate fault if any input/out ports become faulty.

Fault tolerant multistage inter connection networks with widely dispersed paths: Kruskal and Snir (1983) proposed the 2-dilated baseline network is shown in Fig. 5 whose performance in event of fault degrades as gracefully as possible. All the available paths established between an input terminal and an output one via an identical input of a Switching Element (SE) in some stage never pass through an identical SE in the next stage. The loads on SEs, therefore, are shared efficiently. The Extra links added to enhance the performance do not complicate the routing scheme. Besides this MIN is superior to other MIN in performance, especially in robustness against concentrated SE faults in an identical stage.

As shown in below Fig. 5 the paths established between an input terminal and an output one via an identical input of SE in some stage can pass through separate SEs in the next stage A 2 dilated extra link MIN (ELMIN) is proposed by Choi and Somani (1996) subsequently, it is constructed by changing the link connection patterns of first and last stages in 2-dilated MIN. Figure 6 shows a 2-dilated ELMIN with N = 8. In this MIN a path is always established between any input terminal and any output one even if at most four SE faults occurs in the inter-mediate stages. The priority from the first to the fourth is assigned to each

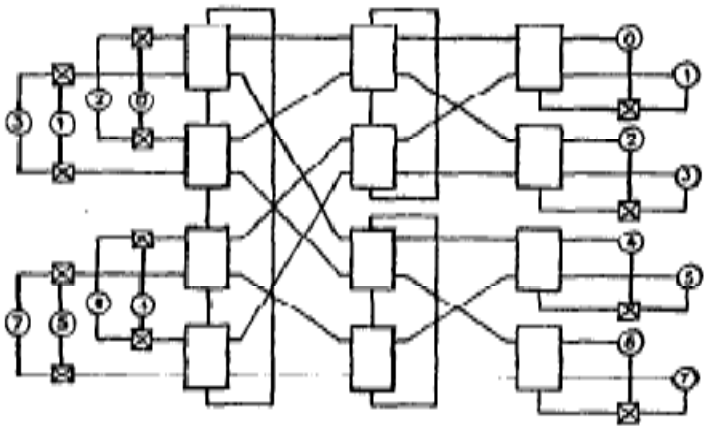


Fig. 4: Fault tolerance in MINs by adding extra switches

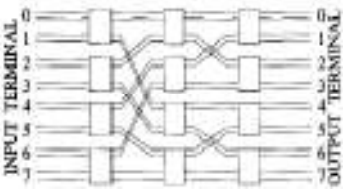


Fig. 5: Illustrates a 2-dilated MIN

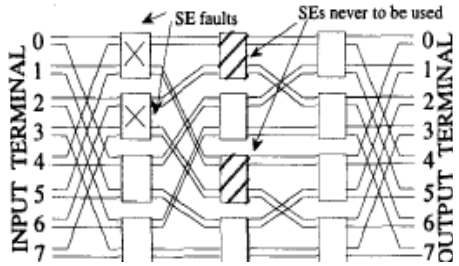


Fig. 6: 2-Dilated ELMIN

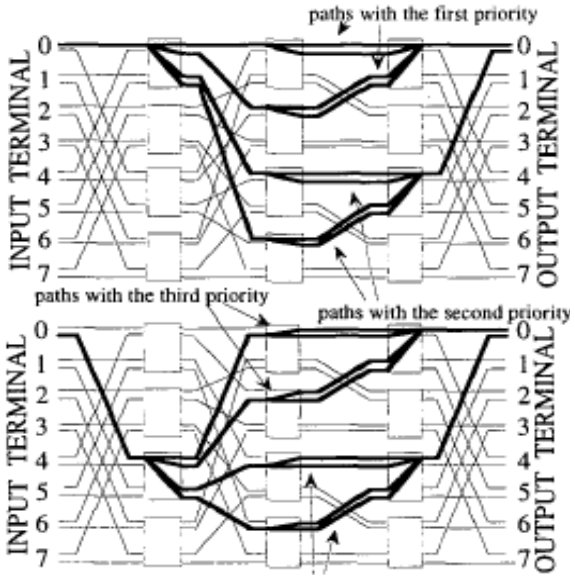


Fig. 7: MIN with N = 8 and paths with priority

available path between any input terminal and any output one.

When some adjacent SE occur in some stage simultaneously, it is possible that some healthy SEs in the next stage can never be used to establish paths. The new MIN proposed by Kamiura *et al*. (2000) consists of multiple paths exist between any source and destination pair and each path is assigned with priority. When any fault link is detected then the path with second highest priority will be chosen. The detail is shown above Fig. 7 which shows multiple paths with priority. In figure four paths establishes in between any source and destination pair. It is possible to established 2^n paths between any input and output terminal.

The black shaded path shows between input port-0 to output port-0. When fault arises in any of above shaded path then the path with less priority will be selected for packet traversal. For example path with first priority is chosen and if fault occur in this path then the path of second priority (next highest) is selected for routing between source to destination pair.

Fault tolerance in MINs with extra hardware: The fault tolerant MINs discussed by Kamiura *et al*. (2000) and Choi and Somani (1996) are less superior than those proposed by Kamiura *et al*. (2002) with respect to throughput and performance. The MIN proposed by Kamiura *et al*. (2002) with N input terminals and N output terminals, switching elements (SEs) in the first and nth stages are duplicated where $n = \log_2 N$ and four-input two-output SEs and two-input four-output SEs are employed in the second and $(n-1)^{th}$ stages, respectively. These extra SEs and links are useful in improving the fault tolerance and performance of the MIN.

Padmanabhan and Lawrie (1983) proposed a MIN with extra stages and Adams and Siegel (1982) incorporated SEs specifically for bypassing faults. These networks usually complicate the routing algorithm or require too much hardware. Choi and Somani (1996) proposed an extra link MIN (ELMIN).

In an ELMIN with N input and N output terminals, the first and nth stages ($n/4 \log_2 N$) consist of four-input two-output SEs and two-input four-output SEs, respectively. It is possible to establish four paths between any input and any output terminal. In their study a MIN is based both on the addition of extra links and on the duplication of SEs. The MIN shown in Fig. 8 corresponds to a hybrid of a non-redundant baseline network and an ELMIN. If the numbers of input and output terminals are N and N respectively, then extra SEs are added to the first and n^{th} stage where $n = \log_2 N$. The link connection pattern between the extra SEs and input (or output) terminals is different to that in a non-redundant baseline network. The Extra links are also added to SEs in the second and (n-1) th stages. In other words, four(or two)-input two(or four)-output SEs in the second or $(n-1)^{th}$ stage are employed.

It can be noted that the choice of SEs at the first stage is independent of the address namely, the routing is also executed according to (0 0010)2 and $(1\ 0\ 0\ 1\ 1)_2$ when we use the fifth SE_2 instead of the first SE_1 in the first stage to establish the path to the output terminal with $(0001)_2$.

However, this MIN can't tolerate two switch faults at either first or last stage where duplicate path from a particular source to the destination covers this two stages. This is the limitation of the MIN (Kamiura *et al*., 2002).

For example as shown in below Fig. 9 four path covers switch number 5 and 1 of stage 1. So if both switches become faulty the path can't be established which creates the bottleneck in the communication.

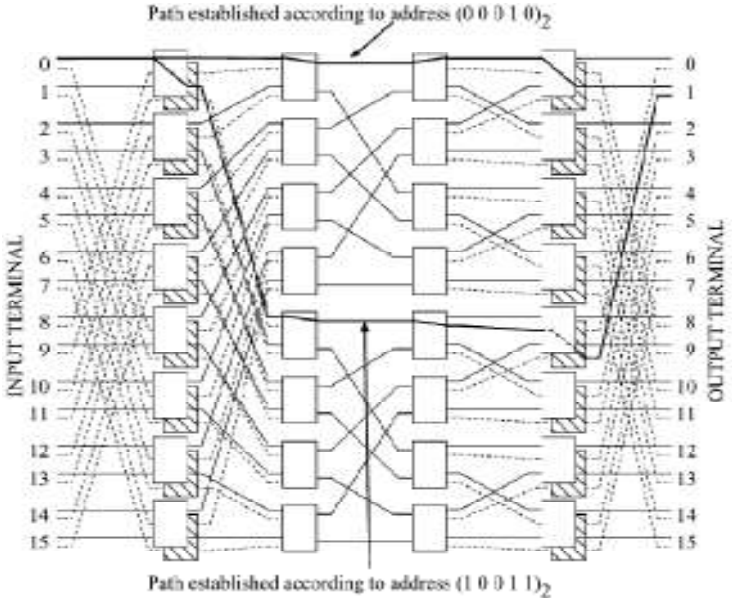


Fig. 8: Illustration of duplicate switch at first and last stage of MIN

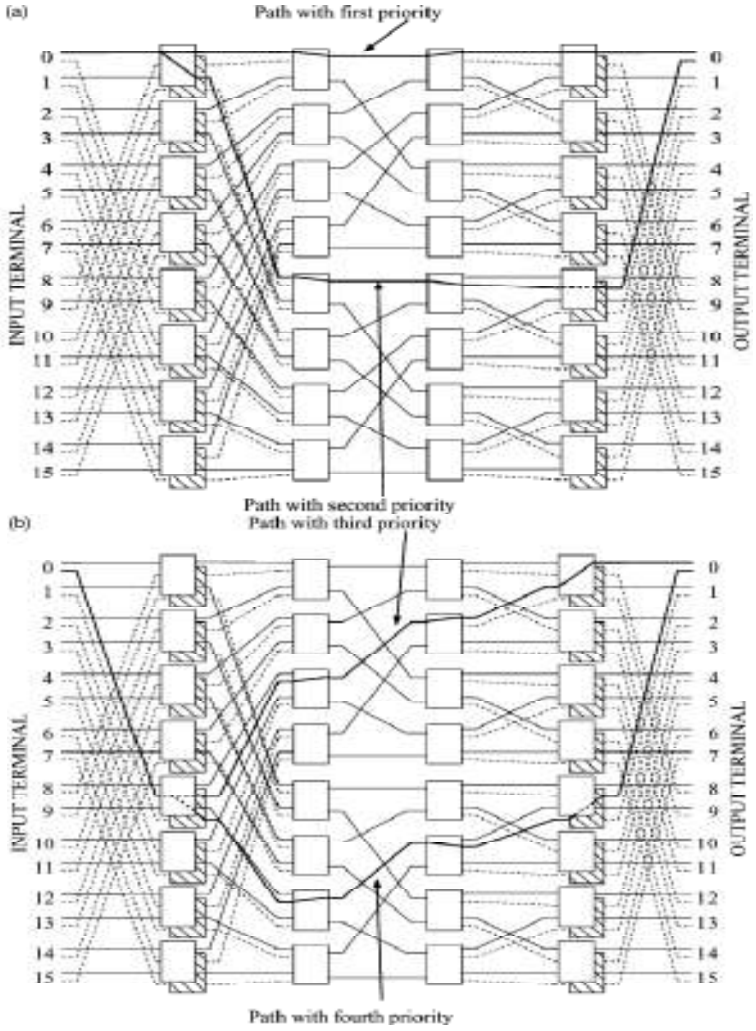


Fig. 9: Path priority; (a): paths with first and second; (b): paths with third and fourth

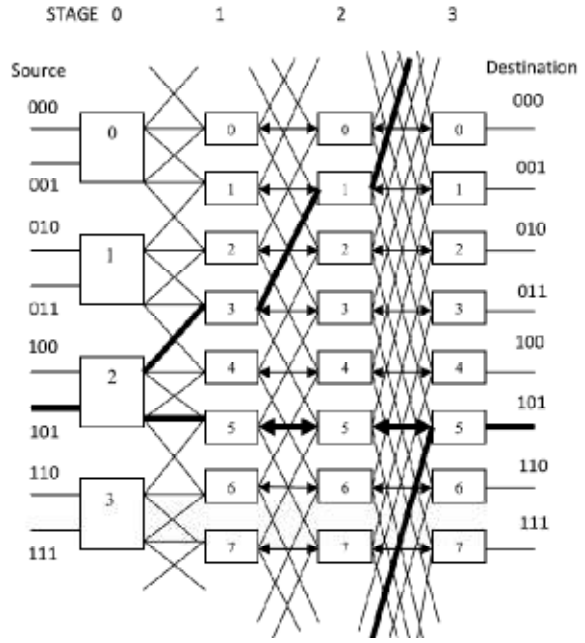


Fig. 10: Combined Switches Multi-stage Interconnection Network (CSMIN)

Fault-tolerance in MINs combining switches with disjoint paths: A fault-tolerant MIN with disjoint path called d Combining Switches Multi-stage Inter connection Network (CSMIN) is proposed by Chen and Chung (2005). The CSMIN is shown in Fig. 10 and it is of size $N = 2^n$ consists of n+1 stages labelled from 0 to n. At stage 0, switch 2i and switch 2i+1 are coupled into a 2×4 switch, for = 0 to (N/2-1). Stage 1 to Stage n have N switches labelled from 0 to 2^n-1. All straight links between stage 1 and stage n are bi-directional. The switch architecture at the first and the last stage has 2×4 and 3×2 crossbars, respectively. Switches located at stage 1 have 3×3 crossbars. Moreover, each switch located at the intermediate stage has a 4×4 crossbar switch. Figure 10 illustrates a CSMIN of size 8.

Subsequently a new design called Fault-tolerant Fully-chained Combing Switches Multistage Interconnection Network (FCSMIN) was proposed Nitin Garhwal and Srivastava (2011) and shown in Fig. 11. The FCSMIN makes use of the destination-tag routing for stages 1 to n to overcome the backtracking problem in CSMIN. The destination-tag routing algorithm does not involve backtracking, it uses bi-directional switches between stages 1 to n and thus bring down the hardware cost of FCSMIN less than CSMIN.

In UpRoute function based FCSMIN, the chaining scheme is that switch j is chained to switch $(j-2^i) \bmod 2^n-1$, where i denotes stage number from 1 to n-1 and $n = \log_2 N$. For example, at stage 1, the chain-out link of switch 2 is connected to the chain-in link of switch 0. In the last stage, remove all the downward (not straight) links are removed.

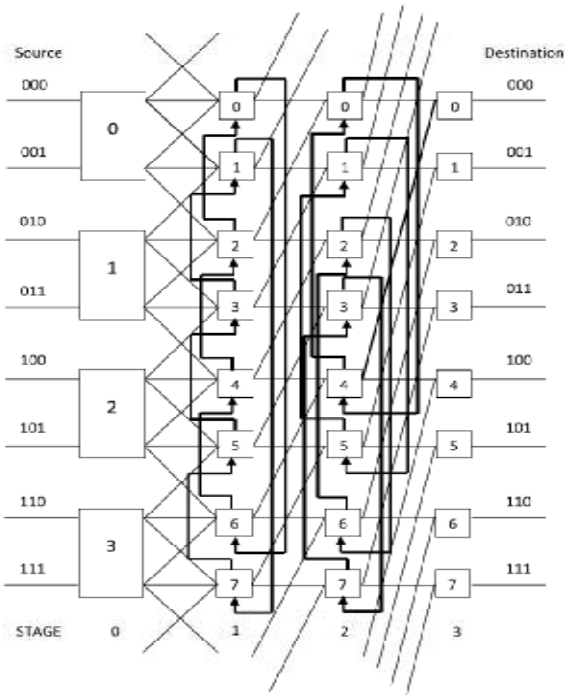


Fig. 11: Illustrates FCSMIN

In DownRoute function based FCSMIN, the chaining scheme is that switch j is chained to switch $(j+2^i) \bmod 2^n-1$, where i denotes stage number from 0 to n-1 and $n = \log_2 N$. For example, at stage 1, the chain-out link of switch 2 is connected to the chain-in link of switch 4. In the last stage, all the upward non straight links have been removed and down route function is:

$$\text{DownRoute}(j, t_i) = \begin{cases} (j+1) \bmod N \text{ at stage } i \\ \quad \text{if } (j_i = 0 \text{ and } t_i = 0) \\ \quad \text{if } (j_i = 1 \text{ and } t_i = 0) \\ (j+1) \bmod N \text{ at stage } i+1. \\ \quad \text{if } (j_i = 0 \text{ and } t_i = 1) \\ \quad \text{if } (j_i = 1 \text{ and } t_i = 1) \end{cases}$$

The up route function is given below for stages 1 to n with chaining links, the routing functions can be derived from the pre-defined UpRoute and DownRoute destination-tag routing functions as:

$$\text{UpRoute}(j, t_i) = \begin{cases} (j-1) \bmod N \text{ at stage } i \\ \quad \text{if } (j_i = 0 \text{ and } t_i = 0) \\ \quad \text{if } (j_i = 1 \text{ and } t_i = 0) \\ (j-1) \bmod N \text{ at stage } i+1 \\ \quad \text{if } (j_i = 0 \text{ and } t_i = 0) \\ \quad \text{if } (j_i = 1 \text{ and } t_i = 0) \end{cases},$$

In CSMIN the fault at first stage and last stage cannot be tolerated so packet will be lost in this case. But in FCSMIN all fault including those at first stage and last stage can also be tolerated.

The purpose of adding multiplexers and demultiplexers at first and last stage of CSMIN are to facilitate fault-tolerance those stages.

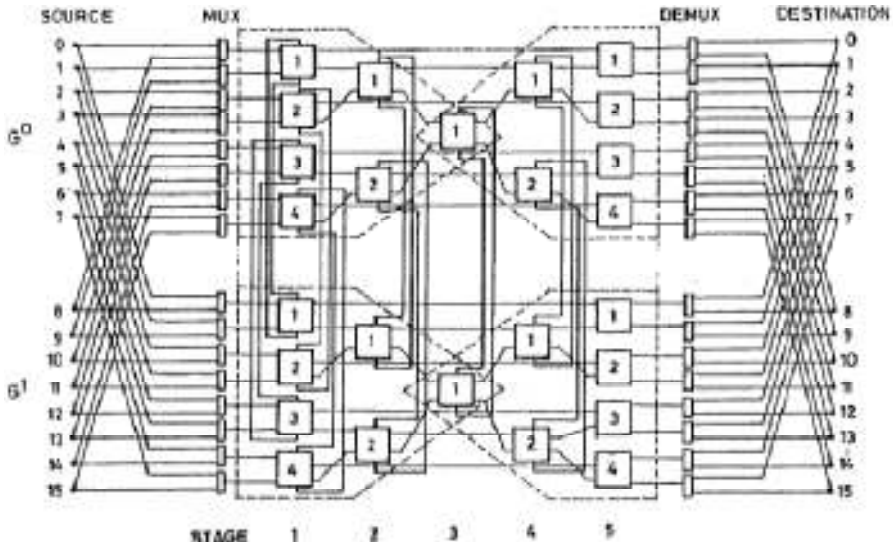


Fig. 12: QT network of 16×16

FAULT-TOLERANCE IRREGULAR MINS

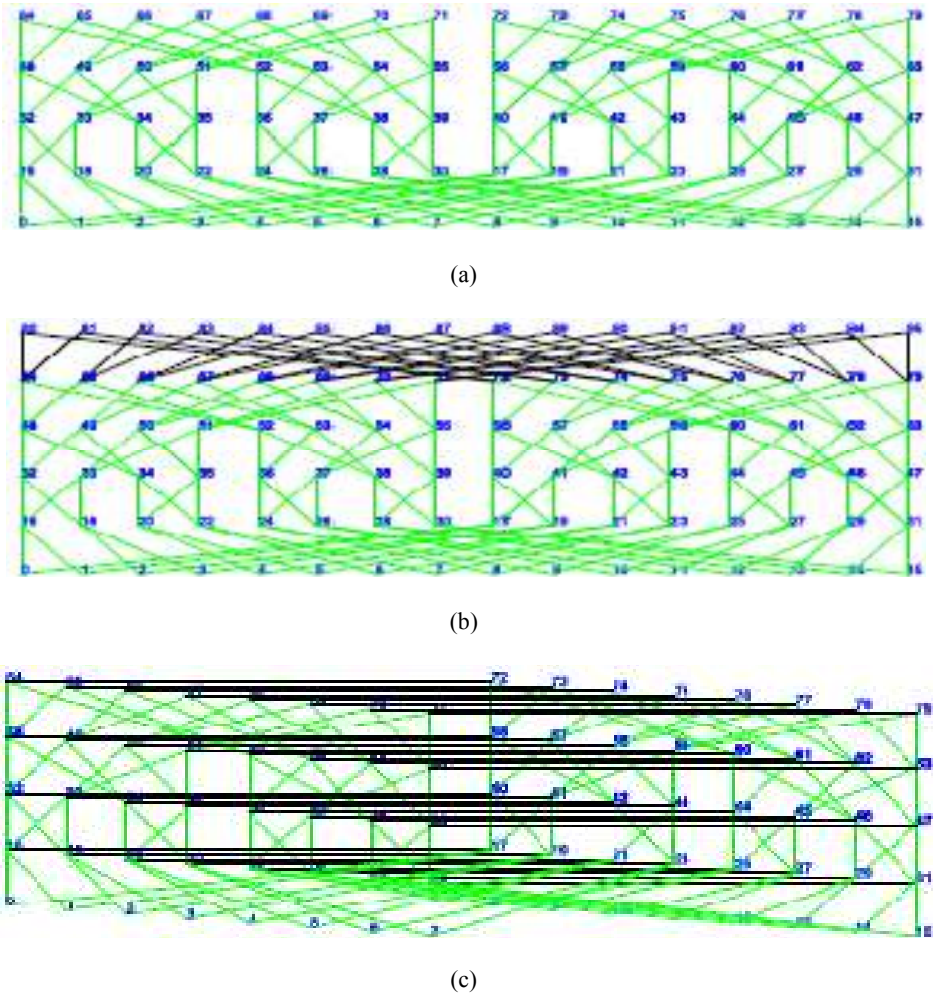
Apart from the regular MINs, there exist some MINs with irregular topologies. Those MINs are referred as irregular topologies MINs (Leiserson, 1985; Sem-Jacobsen *et al*., 2005 and Sengupta and Bansal, 1998). The study of fault tolerance for irregular means is quite essential.

The Quad Tree and Fat Tree and Siamese-twin fat tree are some examples of Irregular MINs studied in the literature. In this section, we studied and reviewed the various means of fault tolerance those are applicable for irregular MINs.

Fault tolerance in (quad tree): The Quad Tree network is a dynamically reroutable irregular MIN that provides multiple path of varying lengths between a Source-Destination pair. This MIN possess Dynamic Full Access (DFA) capability in the presence of multiple faults and is cost effective compared to other fault-tolerant MINs with a similar fault-tolerance capability. The rerouting in the presence of faults can be accomplished dynamically without rerouting to backtracking. The Quad Tree network of size N×N is constructed with two identical groups G', each consisting of MDOT network of size N/2×N/2, which are arranged one above the other ($N = \log_2 N$) is shown in Fig. 12.

The fault-tolerance and performance of this network depends on how effectively the multiple paths are used. Backtracking routing algorithms can be used but the extensive search for the fault-free path can take long time, as also being more expensive. The routing algorithm works quite well. The algorithm assumes that sources and switching elements have the ability to detect faults. The faults in MINs can be detected by the application of test inputs or by employing concurrent error detection at the network or switch level.

Fault tolerance in irregular MINs (fat tree): Fat-trees are a type of irregular MINs which are able to simulate every other network built from the same amount hardware with only small increase in execution time (Bay, 1995). The Fat-trees are therefore well-suited for use in multiprocessor systems to interconnect the processing nodes. The fat-tree topology is similar to ordinary tree topologies, but with one significant difference. Instead of having the tree become thinner nearer the root, the network maintains the high-capacity of the bottom branch level up to the tree root. This gives a tree with higher capacity links nearer the root, or with several roots. The processing nodes are connected to the leaves of the network. The Fat-trees with many roots have good static fault tolerance abilities since the topology provides several alternative paths between every source/destination pair. This requires either a routing algorithm able to adaptively utilise all the paths offered, or the use of a deterministic routing algorithm where the path to be utilised is chosen by the source of a flow. Fat-trees are, however, not able to provide dynamic fault tolerance in their original form. Lysne and Skeie (2001) proposed a modified fat tree which can tolerant fault dynamically and handle faults without halting the network. However, for a large network size with high fault frequency, static fault tolerance is not effective. Further reconfiguration of the network drastically reduces performance. In order to provide dynamic fault tolerance the switches are required to support some sort of escape mechanism allowing packets encountering network faults to dynamically select an alternative path. The high number of paths in multistage interconnection networks such as the fat-tree indicates that they are well suited to provide fault tolerance. The said MINs add a parallel fat tree and create links between corresponding switches in every level of both fat-trees in a



(a)

(b)

(c)

Fig. 13: Shows three topologies; (a): twin; (b): Simple and; (c): Siamese twin

configuration. The new MIN is named as Siamese-Twin fat-tree (ST). In this new MIN the processing nodes are connected to each of the parallel trees through two links to matching switches as shown in Fig. 13c. In the event of a failed link in the downward routing phase, packets may be routed further towards the destination using the crossover path as an escape path. Consequently, a dynamic fault tolerance both in the upward and downward routing phases is achieved. In the fault free case, the parallel networks will double the network capacity assuming a uniform distribution of traffic between the two trees.

The first network topology is called the twin fat tree, a network consisting of two separate fat-trees each with a connection to the processing nodes. In other words, a topology similar to ST, but without the crossover links refer Fig. 13a. The second network is compared with an ordinary fat-tree with the same number of processor connections as the two other topologies. In this case the processing nodes have one link to each of the sub trees in the network Fig. 13b. These three networks have the same basic configuration and the utilisation of the networks is identical in the fault free case. The ST topology does not use its crossover links in the case of no faults and the simple fat-tree topology leaves its topmost switch layer unused in the fault free case. Therefore, all the three topologies behave as the twin fat-tree.

When employing dynamic fault tolerance, ST shows a clear performance improvement over the other two topologies. It was observed that ST provide better fault tolerance than the simple and twin fat-tree topologies. The amount of alternative paths enable this topology with a very good ability to tolerate faults. In the dynamic case, the Siamese Twin fat-tree shows a performance far superior to fat tree and twin fat tree as those not even able to tolerate one single fault. In fact, here dynamic fault tolerance performs only slightly worse than static fault tolerance.

Dynamic fault tolerance in fat trees: The ability of the interconnection network is to maintain a high operational efficiency in presence of faulty components. The fault tolerant capability depends

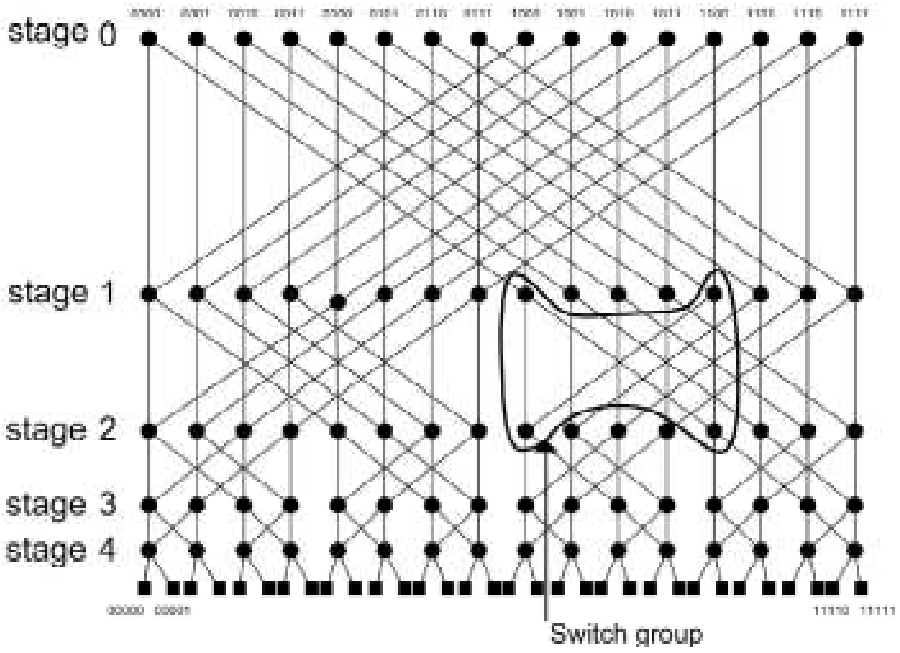


Fig.14: 2- ary-5tree

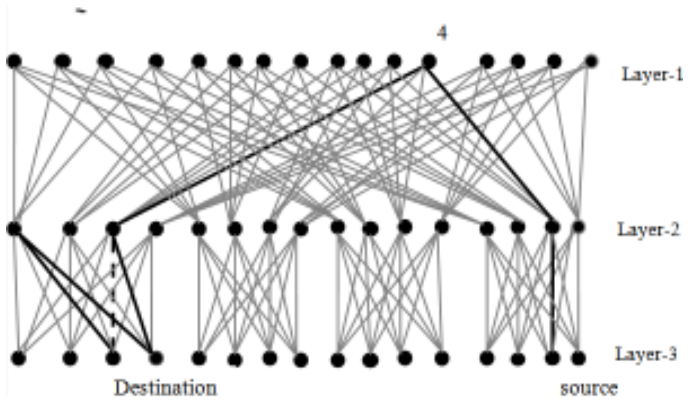


Fig. 15: A fat tree tolerating link fault

strongly on the network topology and the routing function used to generate paths through the network. For the system to remain connected after a fault has occurred there must exist a path between every pair of computing nodes that avoids the failed element. Sem-Jacobsen *et al*. (2011) have proposed a routing method for deterministically and adaptively route in fat trees. It is applicable to both distributed and source routing. This is able to handle several concurrent faults and that transparently returns to the original routing strategy once the faulty components have recovered. The method is local and dynamic. It only requires a small extra functionality in the switches to handle rerouting packets around a fault. The method guarantees connectedness and deadlock and live lock freedom for up to k-1 benign simultaneous switch and/or link faults. Where k is half the number of ports in the switches using either deterministic or adaptive routing where k is half of number of ports of switches. The dynamic local rerouting algorithm also is applicable to source routing for link faults (Sem-Jacobsen *et al*., 2006). A k-ary n-tree is discussed in (Petrini and Vanneschi, 1997) and shown in Fig. 14. It is a k-ary n-tree (for k = 2 and n = 5).

In common for these approaches is that they consider network level fault tolerance based on reconfiguring routing tables. This is achieved either through a central manager instructing the affected nodes to recomputed routing tables, or by permeating updated fault state information through the network from the affected switches (Chen and Chung, 2005). This is time consuming compared to dynamic local rerouting, but later such solutions can be combined with the approaches that are presented by Bay (1995) with a positive result as easy to apply the algorithms.

Figure 15 shows the paths are to be followed when a link is encountered as faulty.

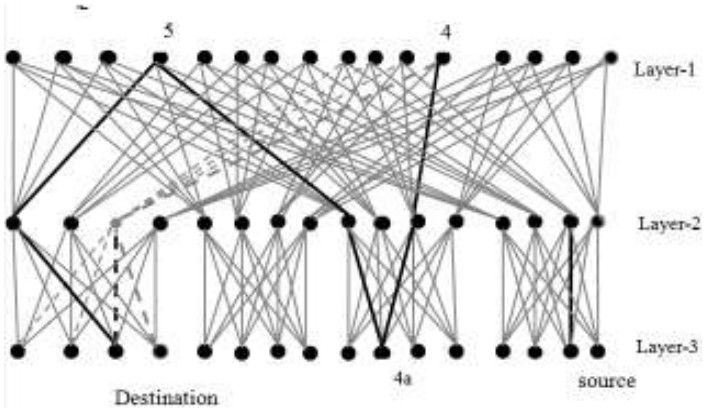


Fig. 16: A fat tree tolerating switch fault.

Figure 15 the dotted line shows a faulty such that packet is rerouted down to leaf and if that leaf is not the destination switch, then it reroutes the packets by U turn towards upward direction. If any downward link in the path is detected as faulty, then it forwards packets in any downward link.

Figure 16 shows paths to be followed when any of these switches become faulty. For the switch-fault tolerance, rerouting down one tier is not sufficient to avoid the faulty switch, as all the paths to a specific destination d within the switch group will lead through the same switchs. However, rerouting down two tiers instead of just one avoids the faulty switch s and achieves connectivity. In this case, it is assumed that the faulty switch s is located at the middle tier of a two-hop switch group G_2

Both of the link faults and the switch faults are tolerated dynamically by local nodes. Both cases follow static and dynamic routing. When there is no fault then it follows static or deterministic routing and if fault occurs in middle of the path then it handle faults dynamically by reroute the packets in alternate path.

FAULT-TOLERANCE IN STATIC INTER-CONNECTION NETWORKS

A static interconnection is a class of interconnection networks which is built out of point to point communication links between processors and memory modules. It is highly suitable for the architectures that consist of large number of homogeneous processors with local memory. It is associated with message passing architecture. Fault tolerance technique in static interconnection networks is highly required. In our literature we have included fault tolerance in interconnection networks based upon combinatorial circuit (Skillicorn, 1988), hyper cube (Leighton, 1992) and Balanced Varietal Hypercube (BVH) (Tripathy and Adhikari, 2011) in next section.

Fault tolerance in ICN based upon combinatorial circuit: Interconnection network based upon

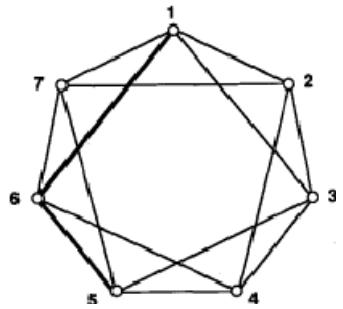


Fig. 17: Shows network based on BIBD with parameter (7, 7, 3, 3, 1)

combinationarial block designs are highly structured and have strong fault tolerant properties. The combinatorial structure is also called as Balanced Incomplete Block Design (BIBD). It contains set of n elements and parameters (n, b, r, k, l). It is a collection of b subsets of size k (called blocks) taken from the set of size n with the property that every distinct element appears precisely one block. The parameter r is called the replication number of design and counts the number of times that each element appears in the collection of blocks. As shown in Fig. 17, each path connects three processors and each processor is connected to four paths. As each processor is having 4 redundant paths so it is obvious that it provides strong fault tolerant capability.

When the failure of any link arises, then the processors need to be informed such that all k processors which are on the path to the failed link belongs get message of link failure.

A processor is notified of a failed link it passes any message that would have used the failed link randomly to one of its neighbour not on the path containing the failure. It tolerate multiple fault with graceful degradation. However this proposed technique is not suitable for multiple switch faults.

Fault-tolerant cycle embedding in static interconnection network: The hypercube is one of the most versatile and efficient static interconnection

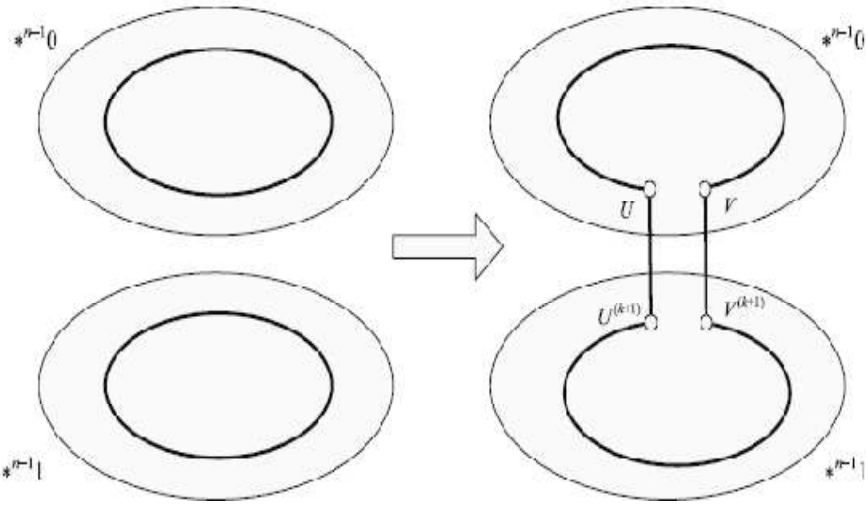


Fig. 18: A basic representation of embedding cycles in cube

networks used parallel computation. It is well suited to both special-purpose and general-purpose tasks and it can efficiently simulate many other networks of the same size. An embedding of one guest graph G into another host graph H is a one-to-one mapping f from the node set of G to the node set of H (Leighton, 1992). An edge of G corresponds to a path of H under f. Fu (2003) has proved that a recursive method of embedding a longest cycle into an n dimensional hypercube which can tolerate 2n-4 faulty nodes. The fault tolerance is more than degree of a node.

A Hamiltonian cycle in a network W is a cycle that contains every node exactly once. Thus, the network W is Hamiltonian if there is a Hamiltonian cycle. The network W is k-link Hamiltonian if it remains Hamiltonian after removing any k links (Harary and Hayes, 1993).The n-dimensional folded hypercube is (n-1) link Hamiltonian (Wang, 2001). The n dimensional star graph is (n-3) link Hamiltonian (Tseng *et al*., 1997). A modification of a d-ary undirected de Bruijn graph is (d-1) link Hamiltonian (Rowley and Bose, 1993). Many results regarding fault-tolerant cycle embedding in a hypercube host graph have been proposed. Latifi *et al*. (1992) showed that the n-dimensional hypercube (n-cube) is (n-2) link Hamiltonian.

A recursive method of embedding cycles in hypercube is shown in Fig. 18. It has been has analytically proved by Fu (2003) that hypercube can tolerate 2n-4 number of node faults where n is the degree of hypercube.

However author it does not mention about the exact or approximate number of link faults that can tolerate.

Fault-tolerance in Balanced Varietal Hypercube (BVH): Tripathy and Adhikari (2011) introduces a new fault tolerant interconnection network topology called

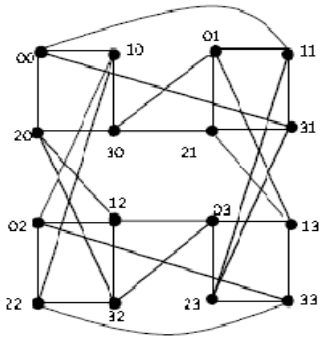


Fig. 19: Balanced varietal hyper cube of dimension-2

Balanced Varietal Hypercube (BVH), suitable for massively parallel systems. The topology being a hybrid structures of Balanced Hypercube and Varietal Hypercube. The performance of the Balanced Varietal Hypercube is compared with Hypercube, Folded hypercube, twisted cube and Crossed cubes. In terms of diameter, cost and average distance and reliability the proposed network is found to be better than the Hypercube, Balanced Hypercube and Varietal Hypercube (Cheng and Chuang, 1994). Also it is more reliable and cost-effective than Hypercube and Balanced Hypercube.

An BVH of n dimension has 2n degree. As shown in Fig. 19 the degree of BVH is four, since four numbers of edges incidents upon a node. The authors of BVH have proved that for any pair of nodes in an n-dimensional Balanced varietal hypercube, there exists 2n disjoint paths between them.

So it can tolerate 2n-1 link faults. When there exist link faults then the alternate link is used for forwarding message. The routing in BVH follows broadcasting of message to all its neighbours. Fault-tolerant routing BVH ensures that message will reach destination if there exist at least one path between source and

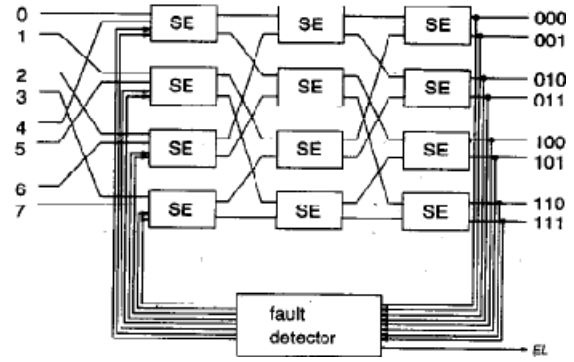


Fig. 20: Fault detection circuit for MINs

destination pair irrespective of number of links or neighbour nodes become faulty. The authors in Wu and Wang (2002) shows better than Hypercube, Varietal hypercube and Balanced hypercube in terms of degree, diameter, cost, average distance and reliability.

FAULT TOLERANT ROUTING IN MINS

An interconnection network may tolerate faults either by adding more hardware components or by rerouting the packets within the network without need of any extra hardware. In next section we discuss it in detail.

Fault tolerant routing in unique path and multipath inter-connection network: Wu and Wang (2002) a routing scheme is described for communication in a multiprocessor system employing a unique-path multistage Inter connection network in the presence of faults in the network. The scheme avoids faulty elements by routing the message to an incorrect destination and then making an extra pass to route to the correct destination. It is capable of tolerating all single fault and many multiple faults in all except the first and last stages of the network. The routing scheme is useful for tolerating both permanent as well as intermittent faults in the network. The technique of tolerating fault in this scheme does not require any extra hardware. So the cost of hardware is less in comparison with Pradhan (1982) where redundant paths are provided by providing extra stage.

The algorithm in Leung (1993) is used for fault diagnosis (detection and location) of baseline ICN in presence of multiple faults. It is based upon number of stages present in ICN. It describes the technique of automatic fault detection. Only the switching element faults can be identified by a circuit i.e., fault detector circuit as shown in Fig. 20.

Figure 20 shows a fault detector circuit connected with L×L switch module. A bit matrix is continuously updated and it keeps track of any faulty switch. It can be implemented by hardware logic circuit.

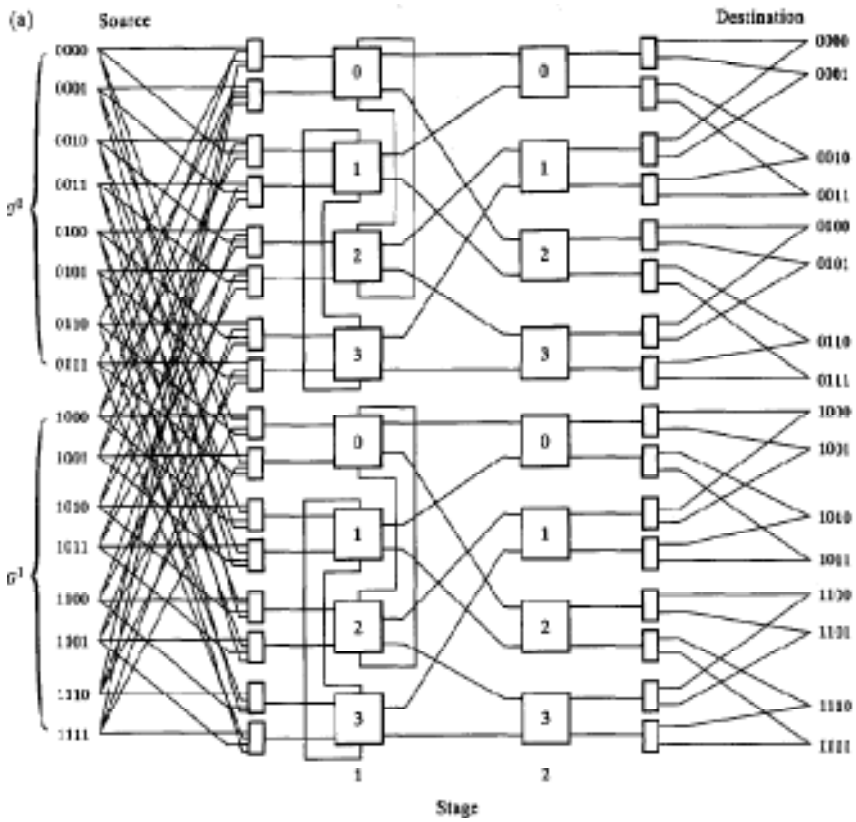


Fig. 21a: Augmented baseline network of size 16

Fig. 21b: Redundancy graph

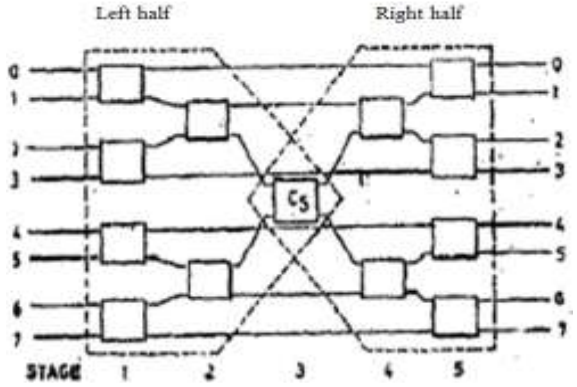


Fig. 22: 8×8 DOT network

Bansal *et al*. (1994) proposed a technique of tolerating fault in new class of multipath ICN named as Augmented Baseline ICNs (ABN) is proposed. The new topology results in a reduced number of stages in the network. The network achieves significant improvements over the unique path MIN (Kim *et al*., 1997). The fault tolerant capability is achieved by creating redundant paths between every source and destination pair. The Augmented Baseline ICNs is shown in Fig. 21a.

The modified baseline network is a network with one less stage and feature like among switches belongs to same stage and forming loops of switches.

The ABN can achieve fault tolerant capability because of the existence of redundant paths in between every source and destination pair as shown in Fig. 21b (redundancy graph).

It can achieve fault tolerant capability with high reliability, good performance even in the presence of faults.

However ABN can tolerate single fault because it maintains two paths in between every source and destination pair namely primary and secondary. In routing, the first primary path is chosen and if found faulty then secondary path is the next alternative. But in case if both the paths become faulty then the network becomes inefficient.

Fault tolerant Compressionless Routing Framework (FCR): The Compression less Routing (CR) is proposed by Kim *et al*. (1997). For adaptive and fault tolerant property. The CR is a framework which provides a unified technique for efficient deadlock free adaptive routing and fault tolerance. The fault tolerance routing supports the end to end fault tolerant delivery. It can be used in most of the interconnection networks. The network interface uses the information to detect possible deadlock situations and network faults and recover from them. The Fault tolerant Compressionless Routing (FCR) extends Compressionless Routing to support end-to-end fault tolerant delivery.

The advantages of Compressionless Routing are:

- Deadlock-free adaptive routing with no virtual channels.
- Simple router designs.
- Order-preserving for message transmission.
- Applicability to a wide variety of network topologies.
- Elimination of the need for buffer allocation messages.

The Compression less Routing, integrates the adaptive routing and fault-tolerant communication. In this framework, possible deadlock situations are detected and recovered without any virtual channels. Thus, CR is compatible with high speed implementations. In addition, Compression less Routing supports fault-tolerant communication under a variety of permanent and transient faults. The performance analysis shows that FCR is performing better than wormhole routing.

Fault tolerant routing in irregular MINs: A simple routing algorithm has been introduced in for two irregular MINs namely Modified fault tolerant double tree (MFDOT) and Quad Tree (QT) where latency and throughput is optimised (Sengupta and Bansal, 1998). Static routing provides full access for MFDOT whereas dynamic routing is provided by QT in presence of faults.

In irregular networks the connection pattern of elements is not uniform from stage to stage so it varies from stage to stage. For non uniform network traffic, an irregular network gives larger throughput than any regular network because of smaller path length. As shown in Fig. 22 the double tree network consists of 8 inputs and 8 outputs. The connection between an input and output pair is set-up by the given. The central switch as shown in Fig. 22 becomes bottleneck in the communication. The central switch is critical and even the presence of a single fault breaks down the system completely.

So, the single central switch is replaced by inter connection of a multiple DOT in MFDOT which becomes fault tolerant because of multiple path formed between every source and destination pair. If any of the switches become faulty, then alternate paths can be chosen. The network MFDOT is shown in Fig. 23.

The 16×16 MFDOT-2 in Fig. 23 provides better fault tolerance to the DOT network. A N×N MFDOT-k is divided into k disjoint sets, Where ($k \geq 2$) and N (>k) are the powers of 2. There are k independent sub networks and an extra one, such that an alternative path is available in the presence of a single fault in the primary module. The MFDOT consists of (2n-1) number stages and (k+1) (2n+1-4) number of switches., where $n = \log_2 N/k$. The MFDOT is associated multiplexers and demultiplexers. It constitute a module, which is denoted as M_0, M_1,..., M_k and equal number of

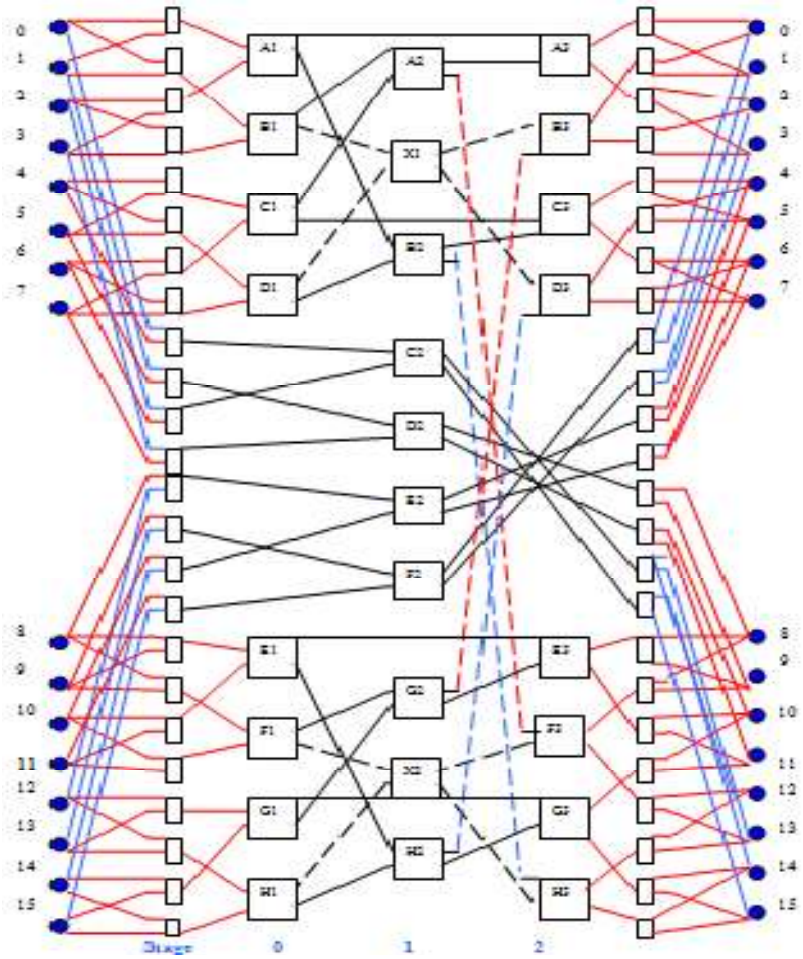


Fig. 23: 16×16 MFDOT-2 network

1×K. Out of multiple paths, the path length algorithm chooses the shortest path which depends on the availability of a fault free path of minimal length.

Fault tolerant and topology flexible routing technique: The fault-tolerant routing in interconnection networks either work for only one given regular topology, or require slow and costly network reconfigurations that do not allow full and continuous network access.

Theiss and Lysne (2006) proposed a routing method for fault tolerance in topology-flexible network technologies. It is based on redundant paths and can handle single dynamic faults without sending control messages other than those that are needed to inform the source nodes of the failing component. In fault-free networks under non uniform traffic, their routing method performs comparable to, or even better than, topology specific routing algorithms in regular networks like meshes and tori.

It is based upon up/down routing which is related to routing in MRoots. Up*/Down* routing (Sancho and Robles, 2000) is a well-known and popular routing algorithm that can be physically adaptive or deterministic.

An Up*/Down* graph is consistent if:

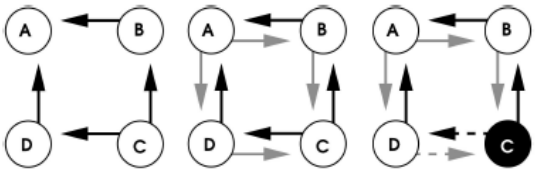
- A node can be chosen to be the root so that there are no cycles consisting of only up-channels or only down-channels in the graph
- The root can be reached from any node following only up-channels
- Any node can be reached from the root by following only down-channels.

All spanning tree channels leading toward the root become up channels and all spanning tree channels leading away from the root become down-channels. The root can be chosen completely randomly, according to ID, or by using a set of heuristics to decide on the "best" root. The spanning tree can be found in several ways, e.g., a Breadth First Search (BFS) or a Depth-First Search (DFS). Figure 24 a as an Up*/Down* graph where node A is the root. The arrows indicate the up-direction of each channel. The network is biconnected, so there are two paths from every source

Table 1: Comparative analysis of fault tolerance in interconnection networks

Interconnection network	Fault model	Types of tolerance (static/ dynamic)	Number of faults (single/multiple)
Baseline ICN	Any components becomes unusable	Static	Multiple
Single stage Beta network	Any components becomes unusable	Static	Single
Augmented Baseline ICN	Any components becomes unusable	Static	Single
CR Routing ICN	Any components becomes unusable	Dynamic	Multiple
MFDOT	Any components becomes unusable	Static	Multiple
QT	Any components becomes unusable	Dynamic	Multiple
2-dilated ELMIN	Any components becomes unusable	Static	Single and limited for multiple
ELMIN with duplicate switch	Any components becomes unusable	Static	Single and limited for multiple faults
Hypercube	Any components becomes unusable	Static	2n-4 faulty node (n is dimension)
Balanced Varietal Hypercube	Any components becomes unusable	Static	Multiple 2n-1 (n is degree)
FROOTS	Any components become unusable	Dynamic	Single
Siamese-Twin fat tree	Any components become unusable	Dynamic	Multiple
FCSMIN	Any components become unusable	Dynamic	Single
FAT TREE	Any components become unusable	Dynamic	Multiple

Interconnection network	Fault tolerance method	Hardware requirements for tolerating fault
Baseline ICN	Alternate route	Extra link required
Single stage Beta network	Through extra pass	No extra hardware required
Augmented Baseline ICN	Alternate route	Extra link added
CR Routing ICN	Through extra pass	Not required
MFDOT	Alternate route	Not required
QT	Alternate route	Not required
2-dilated ELMIN	Alternate route	Extra link is required
ELMIN with duplicate switch	Alternate route	Extra switch
Hypercube	Alternate route	No
Balanced Varietal Hypercube	Alternate route	No
FROOTS	Alternate route	No
Siamese-Twin fat tree	Alternate route	Extra link and switch required
FCSMIN	Alternate route	Extra link and switch required
FAT TREE	Alternate route	No



(a) (b) (c)

Fig. 24: Network with different routing algorithms; The arrow indicate the up-direction of chanels; (a); up*/down*; (b): Redundant routing; (c): FRoots (only two layers shown)

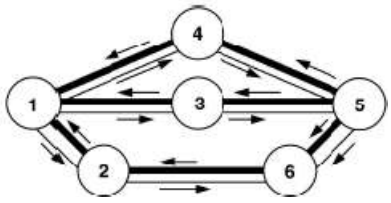


Fig. 25: Network with two roots in two different virtual layers

to every destination, but not two legal paths: packets from B to D have to pass through node A, as do packets from D to B.

In order to guarantee freedom from deadlock, each of these routing functions runs on its own separate set of virtual channels. The nodes injecting packets into the network can decide which set of virtual channels the packet should be routed on (Fig. 25).

In FRoots described the use virtual channels to partition the network into a number of layers. Furthermore, each layer is assigned an individual, deadlock free Up*/Down* graph, in such a way that all nodes are leaves in at least one layer. This allows FRoots to guarantee redundancy for single faults.

In FRoots, the Up*/Down* graphs assigned to each layer are designed to ensure that every node is a leaf in at least one layer. A safe layer of a node is a layer in which the node is a leaf.

If the network has more layers than FRoots needs, it is possible to utilize these layers to increase the number of safe layers of each node. The FRoots can tolerate single fault and DFA property is not discussed so far.

A COMPARATIVE ANALYSIS OF FAULT TOLERANT IN INS

Table 1 summarizes the network fault tolerance information presented in our survey. It lists: the possible faults that can occur in each network under the assumed fault model; whether or not faulty components are usable; the fault-tolerance criterion; the method by which the network copes with faults; whether the network is single-fault tolerant; and how the network performs with multiple faults.

In case of multiple fault most of network is limited by the number of switch or link faults.

For example in ELMIN four paths are exist between every source and destination pair. So it can

tolerate 3 numbers of faulty links between any source and destination pair. But if two switches where all four paths are passes become faulty then source communication between particular source destinations becomes impossible. In fault model any component can become faulty. Many of the networks fail to be single fault tolerant because they cannot tolerate an input or output switch fault. Thus many fault models refer only to interior switch faults.

CONCLUSION

We compared and surveyed the fault tolerance interconnection networks. This tolerance can be achieved by modifying the network by either adding extra link or switch. Some of the methods only change the routing technique of message without extra hardware. We have included most of networks varies from single stage to multistage interconnection network. Besides the regular topology, irregular topology interconnection networks have been included in our survey. The fault tolerant routings may handle faults dynamically or statically also included in detail.

REFERENCES

Adams, G.B. III and H.J. Siegel, 1982. The extra stage cube: A fault-tolerant interconnection network for supercomputer systems. IEEE T. Comput., C-31: 443-454.

Adams III, G.B., D.P. Agrawal and H.J. Siegal, 1987. A survey and comparision of fault-tolerant multistage interconnection networks. Computer, 20: 14-27.

Bansal, P.K., R.C. Joshi and K. Singh, 1994. On a fault-tolerant multi-stage interconnection network. Int. J. Electron. Electr. Eng., 20(4): 335-345.

Bay, P., 1995. Deterministic online routing on area-universal networks. J. Assoc. Comput. Mach., 42(3): 614-640.

Chen, C.W. and C.P. Chung, 2005. Designing a disjoint path interconnection network with collision solving and fault tolerance. J. Supercomput., 34(1): 63-80.

Cheng, S.Y. and J.H. Chuang, 1994. Varietal hypercube-a new interconnection network topology for large scale multicomputer. Proceedings of International Conference on Parallel and Distributed System, pp: 703-708.

Choi, S.B. and A.K. Somani, 1996. Design and performance analysis of load-distributing fault-tolerant network. IEEE T. Comput., 45(5): 540-551.

Dash, R.K., N.K. Barpanda, P.K. Tripathy and C.R. Tripathy, 2012. Network reliability optimization problem of interconnection network under node-edge failure model. Appl. Soft Comput., 8: 2322-2328.

Feng, T.Y., 1981. A survey of interconnection networks. Computer, 14: 12-17.

Fu, J.S., 2003. Fault-tolerant cycle embedding in the hypercube. Parallel Comput., 29: 821-832.

Harary, F. and J.P. Hayes, 1993. Edge fault tolerance in graphs. Networks, 23: 135-142.

Huang, C.F. and W.T. Chen, 1987. Fault-tolerant single-stage interconnectiohl networks. IEEE T. Comput., C-36(5): 637-640.

Kamiura, N., T. Kodera and N. Matsui, 2000. Fault tolerant multistage interconnection networks with widely dispersed paths. Proceedings of the Ninth Asian Test Symposium, pp: 423-428.

Kamiura, N., T. Kodera and N. Matsui, 2002. A fault tolerant multistage interconnection network with partly duplicated switches. J. Syst. Architect., 47: 901-916.

Kim, J.H., L. Ziqiang and A. Chien, 1997. Compressionless routing*:* A framework for adaptive and fault-tolerant routing. IEEE T. Parall. Distr., 8(30): 229-243.

Kruskal, C.P. and M. Snir, 1983. Performance of multi stage interconnection networks for multi processors. IEEE T. Comput., c-32(12): 1091-1098.

Latifi, S., S.Q. Zheng and N. Bagherzadeh, 1992. Optimal ring embedding in hypercube with faulty links. Proceeding of the IEEE Symposium on Fault-tolerant Computing, pp: 178-184.

Leighton, F.T., 1992. Introduction to Parallel Algorithms and Architecture: Arrays. Trees, Hyper Cubes, Morgan Kaufman, CA.

Leiserson, C.E., 1985. Fat-trees: Universal networks hardware efficient supercomputing. IEEE T. Comput., 34(10): 892-901.

Leung, Y.W., 1993. Online fault identification in multistage interconnection networks. Parallel Comput., 19: 693-702.

Lysne, O. and T. Skeie, 2001. Load balancing of irregular system area networks through multiple roots. Proceeding of 2nd International Conference on Communications in Computing, 26: 62-76.

Nitin Garhwal, S. and N. Srivastava, 2011. Designing a fault-tolerant fully-chained combining switches multi-stage interconnection network with disjoint paths. J. Supercomput., 55: 400-431.

Padmanabhan, K. and D.H. Lawrie, 1983. A class of redundant path multistage interconnection networks. IEEE T. Comput., 32(12): 1099-1108.

Petrini, F. and M. Vanneschi, 1997. K-ary N-trees: High performance networks for massively parallel architectures. Proceeding of 11th International Symposium Parallel Processing (IPPS '97), pp: 87.

Pradhan, D.K., 1982. On a class fault-tolerant multiprocessor network architectures. Proceeding of 3rd International Conference on Distributed Computing System, pp: 302-311.

Rowley, R.A. and B. Bose, 1993. Fault-tolerant ring embedding in de Bruijn networks. IEEE T. Comput., 42(12): 1480-1486.

Sancho, J.C. and A. Robles, 2000. Improving the up*/down* routing scheme for networks of workstations. Proceeding of the 6th International Euro-Par Conference, pp: 882-889.

Sem-Jacobsen, F.O., T. Skeie, O. Lysne,O. Tørudbakken, E. Rongved and B. Johnsen, 2005. Siamese-twin: A dynamically fault tolerant fat tree. Proceeding of the 19th IEEE International Parallel and Distributed Processing Symposium.

Sem-Jacobsen, F.O., O. Lysne and T. Skeie, 2006. Combining source routing and dynamic fault tolerance. Proceedings of 18th International Symposium on Computer Architecture and High Performance Computing (SBACPAD), pp: 151-158.

Sem-Jacobsen, F.O., T. Skeie, O. Lysne and J. Duato, 2011. Dynamic fault tolerance in fat trees. IEEE T. Comput., 60(40): 508-524.

Sengupta, J. and P.K. Bansal, 1998. Fault-tolerant routing in irregular MINs. Proceeding of the IEEE Region 10 International Conference on Global Connectivity in Energy, Computer, Communication and Control, 2: 638-641.

Shen, J.P. and J.P. Hayes, 1984. Fault-tolerance of dynamic-full-access interconnection networks. IEEE T. Comput., C-33: 241-248.

Skillicorn, D.B., 1988. A new class of fault-tolerant static interconnection networks. IEEE T. Comput., 31(11): 1468-1470.

Street, A.P. and W.D. Wallis, 1977. Combinatorial Theory: An Introduction. Charles Babbage Research Centre, Winnipeg, Canada.

Theiss, I. and O. Lysne, 2006. FRoots: A fault tolerant and topology-flexible routing technique. IEEE T. Parall. Distr., 17(10).

Tripathy, C.R. and N. Adhikari, 2011. A new multicomputer interconnection topology for massively parallel systems. Int. J. Distrib. Parallel Syst. (IJDPS), 2(4).

Tseng, Y.C., S.H. Chang and J.P. Sheu, 1997. Fault-tolerant ring embedding in a star graph with both link and node failures. IEEE T. Parall. Distr., 8(12): 1185-1195.

Tzeng, N.F., P.C. Yew and C.Q. Zhu, 1985. A fault tolerant scheme for multistage interconnection network. Proceeding of International Conference on Parallel Processing, pp: 368-375.

Wang, D.J., 2001. Embedding Hamiltonian cycles into folded hyper cubes with link faults. J. Parall. Distr. Com., 61(4): 545-564.

Wu, J. and D. Wang, 2002. Fault-tolerant and deadlock-free routing in 2-D meshes using rectilinear-monotone polygonal fault blocks. Proceeding of International Conference on Parallel Processing, pp: 247-254.