Fault-Tolerance and Permutation Analysis of ASEN and its Variant

Rinkle Rani Aggarwal[1], Dr. Lakhwinder Kaur[2]
Department of Computer Science & Engineering,
Thapar University,
Patiala –147004 (INDIA)
raggarwal@thapar.edu

Department of Computer Engineering,
Punjabi University,
Patiala–147002 (INDIA)
mahal2k8@yahoo.com

***Abstract-* High performance computing systems can be designed using parallel processing. The effectiveness of these parallel systems rests primarily on the communication network linking processors and memory modules. Hence, an interconnection network that provides the desired connectivity and performance at minimum cost is required. The design of a suitable interconnection network for inter-processor communication is one of the key issues of the system performance. In this paper a new multistage interconnection network IASEN (Irregular augmented shuffle exchange) has been proposed modifying existing ASEN-2 network. ASEN-2 is a regular multipath network with limited fault tolerance. The paper also discusses the permutation possibility behavior of both the networks with and without faults. It has been observed that the proposed multistage interconnection network IASEN provides much better fault-tolerance by providing more paths between any pair of source-destination as compared to ASEN-2 at the expanse of little more cost.**

***Keywords-* Multistage Interconnection Networks, Bandwidth, Augmented Shuffle Exchange Network, Irregular Augmented Shuffle Exchange Network, Fault-tolerance, Permutation.**

I. INTRODUCTION

Today is the era of parallel processing and building of multiprocessor system with hundred processors is feasible. Interconnection Networks (INs) play a major role in the performance of modem parallel computers. A vital component of these systems is the Interconnection Network (IN) that enables the processor to communicate themselves or with memory units [12]. Many aspects of INs, such as implementation complexity, routing algorithms, performance evaluation, fault-tolerance, and reliability have been the subjects of research over the years. There are many factors that may affect the choice of appropriate interconnection network for the underlying parallel computing environment. Though crossbar is the ideal IN for shared memory multiprocessor, where N inputs can simultaneously get connected to N outputs, but the hardware cost grows astronomically [6]. Multistage Interconnection Networks (MINs) are recognized as cost-effective means to provide programmable data paths between functional modules in multiprocessor systems [9]. These networks are usually implemented with simple modular switches, employing two-input two-output switching elements. Most of the MINs proposed in the literature have been constructed with 2x2 crossbar switches as basic elements, and have $n = \log_2 N$ switching stages with each stage consisting of N/2 elements, which makes the cost of this network as O(NlogN), as compared to $O(N^2)$ for a crossbar [4]. The pattern of interconnection may be uniform or non-uniform, which classifies the MINs to be regular or irregular respectively [7]. In the case of irregular networks, the path length varies from any input to any output, in contrast with regular networks, where it is the same. Today INs are used in a variety of applications such as switch and router fabrics, processor memory interconnect, I/O interconnect and on-chip networks etc. [8]. Fault-tolerance in an interconnection network is very important for its continuous operation over a relatively long period of time [10]. Fault-tolerance is the ability of the network to operate even in the presence of faults, although at a degraded performance. Many networks have been designed and proposed to increase the fault-tolerance in the literature [1,4,6,10,11,14]. Reliability and other performance parameters have also been studied in depth [3,4,10,12,15,17] but little attention has been paid to the permutation of these networks. This paper has been organized into five sections whose details are as follows.

Section I introduces the subject under study. Section II describes the structure and design of networks. Section III focuses on the redundancy graphs and routing scheme. Section IV concentrates on the permutation possibility behavior of MINs. Section V gives the cost effectiveness. Finally, the conclusions have been presented in Section VI.

II. STRUCTURE AND DESIGN OF NETWORKS

A. *ASEN-2 Network*

Augmented Shuffle Exchange Network (ASEN-2) is a regular network, means it has same number of switches in each stage. ASEN-2 network is constructed from Shuffle Exchange network by adding a stage of 2×1 multiplexers at the initial stage and 1×2 demultiplexer at last stage. It provides multiple paths between a source and a destination.

ASEN-2 of size N × N with N number of sources and N number of destination consists of $\log_2 N$ -1 stages where the initial stage consists of N/2 switches of size 3×3 and the last stage consist of N/2 switches of size 2×2 [14]. In each stage, the switches can be grouped into conjugate subsets, where a conjugate subset is composed of all switches in a particular stage that lead to the same subset of destinations. The switches, which communicate through the use of auxiliary links, form a conjugate loop. The conjugate loops are formed in such a way that the two switches that form a loop have their respective conjugate switches in a different loop. This pair of loops is called conjugate loops [5,6]. ASEN-2 provides fault tolerance using links between the conjugate pairs of switches. Fig. 1 shows an ASEN-2 of size 16x16.

B. Irregular ASEN

Irregular Augmented Shuffle Exchange Network (IASEN) is derived from ASEN-2 multistage interconnection network (Figure 2). An N × N ($2^n \times 2^n$) network consists of m stages (where m = $\log_2$ N/2). The first and the last stage of the network consist of equal number of switching elements that is 2^{n-1} each whereas the intermediate stages consist of less number of switching elements equal to 2^{n-2} each. The switches in the last stage are of size 2 × 2 whereas stages from 1 to m-1 are having switches of size 3 × 3. Thus, the total number of switches are equal to $2^{n-2}(m+2)$ out of which 2^n number of switches are of size 2 × 2 and (m-2) × 2^{n-2} number of switches are of size 3 × 3. The switches in the first stage form a loop to provide multiple paths if a fault occur in the next stage. Each source is connected to two different switches in each group with the help of multiplexer and each destination is connected with demultiplexer. Following structural changes have been made in IASEN in comparison to ASEN-2.

1) Four switches removed from the stage 1 (Intermediate Stage).
2) Use of 1 x 4 DEMUX in place of 1 x 2 DEMUX.
3) Loops and connections changed.

Fig. 2 shows an IASEN of size 16x16.

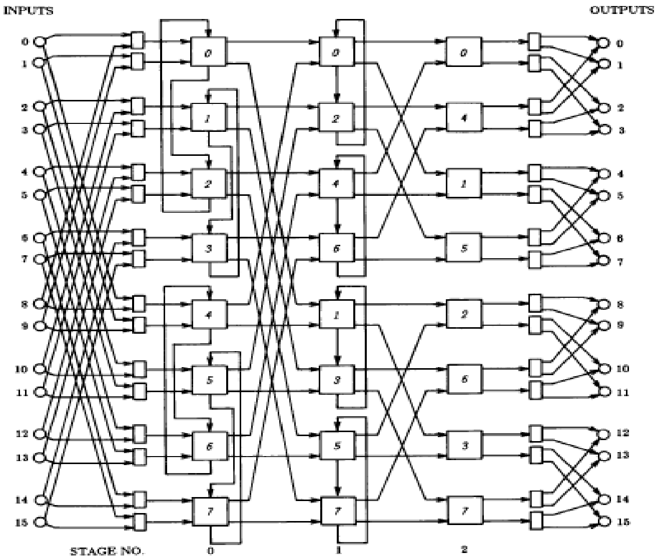

Fig. 1 ASEN-2 of 16x16

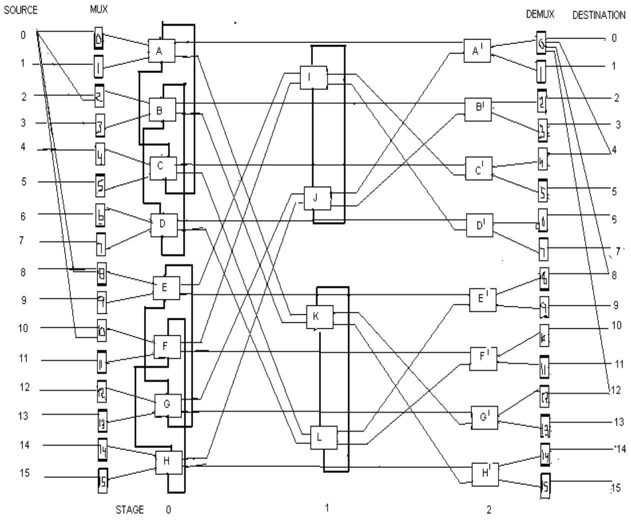

Fig. 2 Irregular ASEN of 16x16.

III. ROUTING SCHEME

A redundancy graph offers a convenient way to study the properties of a multi-path MIN, such as the number of faults tolerated or the type of rerouting possible. A redundancy graph depicts all the available paths between a source and a destination in a MIN [3]. It consists of two distinguished nodes-the source S and the destination D-and the rest of the nodes correspond to the switches that lie along the paths between S and D. The redundancy graph of ASEN-2 shows that there exist six paths between any source and destination, as shown in figure 3. Whereas in case of IASEN there exist eight paths between a source-destination pair, as shown in Fig. 4.

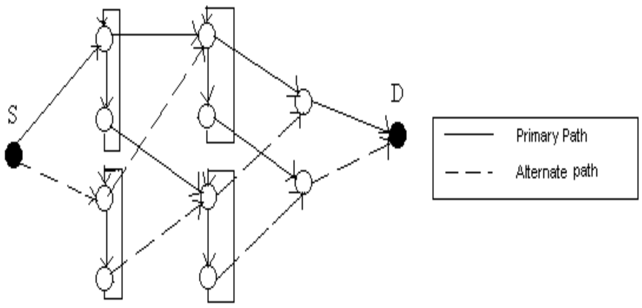

Fig. 3 Redundancy Graph of ASEN-2

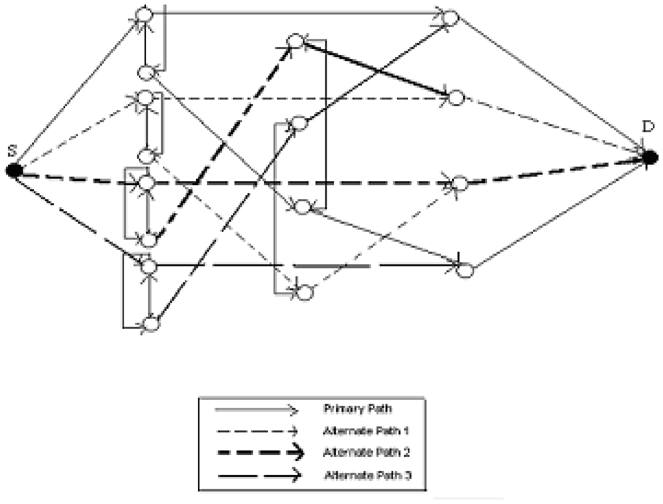

Fig. 4 Redundancy Graph of IASEN

Routing tag in a network is used to describe the path between a source destination pair through the network. Let the source S and destination D be represented in binary code as:

$$S=s_0, s_1\ldots s_{n-2}, s_{n-1}$$
$$D=d_0, d_1\ldots d_{n-2}, d_{n-1}$$

A. *Routing procedure for ASEN-2*

- For each source submit the request to the primary switch, along with the binary label (d_0) of the required destination. If the primary switch, or the multiplexer at the input of the primary switch, is faulty, then submit the request to the secondary switch. If the secondary switch is also faulty then drop the request.
- For each switch in initial stage requests may arrive on any of the 3 input links. For each request if the required output link is busy or cannot be used because of a fault in the next stage, route the request via the auxiliary output link to the next switch in the loop. If the auxiliary link is also unusable because it is busy or faulty, then drop the request.
- For each switch in intermediate stage requests may arrive on any of three input links, as there are 3 × 3 switches are used. For each request, route it via corresponding output link if it is not available then submit the request to the alternate output link. If the alternate link is faulty or busy then drop the request.
- For each switch in last stage requests may arrive on any of two input links, as 2×2 switches are there. For each request route it via the corresponding output link. If the required output link is busy or faulty then drop the request.
- Each demultiplexer can receive maximum of one request. If a request arrives make a connection to the upper or the lower output link according to whether the routing bit is 0 or 1. For each destination up to two requests may arrive [15].

B. *Routing procedure for IASEN*

- For each source submit the request for connection to the primary switch. If the primary switch, or the multiplexer at the input of the primary switch, is faulty or busy, then submit the request to the alternate paths. If no path is available then drop the request.
- For each switch in initial stage requests for connection can arrive on any of the three input links. If the required output link is busy or faulty in the next stage, route the request via the alternate paths, if alternate paths are not available then drop the request.
- For each switch in intermediate stage requests can arrive on any of three input links. If the corresponding output link is busy or faulty then submit the request to the alternate output link. If the alternate link is faulty or busy drop the request.
- For each switch in last stage requests can arrive on any of two input links. If the required output link is busy or faulty then drop the request.
- For each demultiplexer can receive a maximum of one request. If a request arrives, make a connection to any output link according to the routing tag. For each destination up to four requests can arrive as there are 1× 4 demultiplexers are used.

Following algorithms are used for the selection of primary or secondary path and generating the routing tag.

Path length algorithm

If (S is even)

Then

```
    If
        (S = D or D = S+1 or S-D = ± 4 or S- D = ± 8
         or S-D = ± 12)
    Then
            Path length is minimum.
Else
    If
        (S is odd)

    Then
            If   (S = D or D = S-1 or S-D = ± 4 or S-D
                     = ± 8 or S-D = ± 12)
            Then
                  Path length is minimum.
    Else
            The longest path is possible only.
```

Routing algorithm

```
 If
        (path length = minimum)

    then

        Routing tag = $0.D_1.D_{n-1}$
Else

    Set $D_1$=1
    Routing tag = $D_0.D_1.D_2.D_{n-1}$
```

Example:

Let source is 0 and destination is 4

Therefore,
S=0000 and D=0100

Following eight paths are possible from S to D as shown in figure 5.

Primary path:

0->MUX (0)->A->A'->DEMUX (0) ->4

Alternate Paths :

0->MUX (0)->A->C->C'->DEMUX (4) ->4

0->MUX (2)->B->K->G'->DEMUX (12) ->4

0->MUX (2)->B->D->L->E'->DEMUX (8) ->4

0->MUX(8)->E->E'->DEMUX(8)->4

0->MUX (8) ->E->G->G'->DEMUX (12) ->4

0->MUX (10) ->F->I->C'->DEMUX (4) ->4

0->MUX (10) ->F->H->J->A'->DEMUX (0) ->0

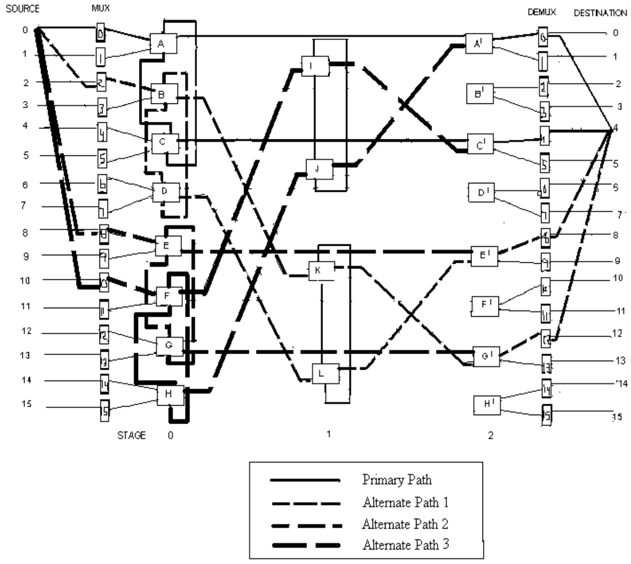

Fig. 5 Alternate paths in IASEN

IV. PERMUTATION PASSIBILITY BEHAVIOR OF ASEN-2 AND IASEN

A. Permutation in absence of faults

This section describes the permutation passibility of two networks ASEN-2 and IASEN in absence of faults.

i. ASEN-2 Network

CASE I: Total number of requests: 6, Number of faults: 0

Let the six source destination pairs are (4,2), (1,3), (9,6), (3,10), (8,2), (7,10). Paths followed by this set of requests are:

[4,2] = 4 → MUX(4)→C→C1→A'→DEMUX(1) → 2

[1,3] = 1→ MUX(1) →A→A1→B1→B'→MUX(3)→3

[9,6] = 9→ MUX(9)→E→A1→C'→DEMUX(5)→6

[3,10] = 3 → MUX(3) →B→ F1→ F → DEMUX(11)→10

[8,2] = 8 → MUX(8) → E→G→C1→A'→DEMUX(1)→2

[7,10] = 7→ MUX(15)→ H→ H1→G1→E'→ DEMUX(9) →10

Number of requests matured successfully: 6
Average Path Length: = (3+4+3+3+4+4)/6
=2.666666

CASE II : Total number of requests : 9, Number of faults : 0

Let the nine source destination pairs are (1,3), (2,4), (3,5), (6,5), (12,4), (13,2), (9,6), (4,12), (5,0). Paths followed by this set of requests are:

[1,3] = 1→MUX(1) → A →A1→A'→DEMUX(1) →3

[2,4] = 2→MUX(2) →B→B1→D'→DEMUX(6) →4

[3,5] = 3→MUX(11) →F→B1→A1→C'→DEMUX(4) →5

[6,5] = 6→MUX(6) →D→D1→D'→DEMUX(6) →5

[12,4] = 12→MUX(4) →C→C1→C'→DEMUX(4) →4

[13,2] = 13→MUX(13) →CLASH

[9,6] = 9→MUX(9) →E→A1→C'→DEMUX(5) →6

[4,12] = 4→MUX(12) →G→G1→G'→DEMUX(12) →12

[5,0] = 5→MUX(5) →CLASH

Number of requests matured successfully: 7
Average Path Length:
= (3+3+4+3+3+0+3+3+0) / 7 = 3.1428

CASE III: Total number of requests: 11, Number of faults : 0

Let the 11 source destination pairs are (0,1), (4,5), (6,2), (7,8), (9,10), (12,14), (3,5), (1,11), (2,5), (15,7), (14,9). Paths followed by this set of requests are:

[0,1] = 0→ MUX(0)→A→A1→A'→DEMUX(0)→1

[4,5] = 4→MUX(4)→C→C1→C'→DEMUX(4)→5

[6,2] = 6 →MUX(6)→ D→ D1→B' →DEMUX(3)→2

[7,8] = 7 →MUX(15)→H→H1→F'→DEMUX(10)→8

[9,10] = 9→MUX(9)→E→E1→E'→DEMUX(8)→10

[12,14]= 12→MUX(12)→G→G1→G'→DEMUX(13) →14

[3,5] = 3→MUX(3)→B→B1→D'→DEMUX(6)→5

[1,11] =1→MUX(1)→A→C→G1→E'→DEMUX(9)→ 11

[2,5] = 2→MUX(2)→B→D→D1→D'→DEMUX(6)→5

[15,7] = 15 → CLASH

[14,9] = 14→MUX(14)→H→F→F1→F→DEMUX(10) → 9

Number of requests matured successfully: 10
Average Path Length : (3+3+3+3+3+3+3+4+4+0+4) /10 =3.3

CASE IV: Total number of requests: 8, Number of faults : 0

Let the 8 source destination pairs are (1,3), (2,4), (3,5), (6,5), (9,6), (4,12), (5,14), (7,9). Paths followed by this set of requests are:

[1,3] = 1→ MUX(1) → A → A1 →A'→DEMUX(1) → 3

[2,4] = 2 → MUX(2)→B→ B1 →D' → DEMUX(6)→ 4

[3,5] = 3 →MUX(3)→B→D→D1→D'→DEMUX(6) →5

[6,5] = 6 → MUX(14) → CLASH

[9,6] = 9 → MUX(9)→E→A1→C'→DEMUX(5)→6

[4,12]=4 →MUX(12)→G→G1→G'→DEMUX(12)→ 12

[5,14]= 5 → MUX(5) →C→F1→ H' → DEMUX(15) → 14

[7,9] = 7 → MUX(15) → H→ H1→ F'→ DEMUX(10)→ 9

Number of requests matured successfully: 7
Average Path Length
=(3+3+4+0+3+3+3+3) / 7 = 3.1428

ii. IASEN Network

CASE I: Total number of requests: 6, Number of faults: 0

Let the six source destination pairs are (4,2), (1,3), (9,6), (3,10), (8,2), (7,10). Paths followed by this set of requests are:

[4,2] = 4→MUX(4) →C→L→F'→DEMUX(10) →2

[1,3] = 1→MUX(1) →A→K→H'→DEMUX(15) → 3

[9,6] = 9→MUX(3) →B→D→L→F'→DEMUX(10) →6

[3,10] = 3→MUX(11) →F→F'→DEMUX(10) →10

[8,2] = 8→MUX(2) →B→B'→DEMUX(2) →2

[7,10] = 7→MUX(15) →H→H'→DEMUX(14) →10

Number of requests matured successfully : 6
Average Path Length:
=(3+3+4+2+2+2) / 6 = 2.666666

CASE II: Total number of requests: 9, Number of faults: 0

Let the nine source destination pairs are (1,3), (2,4), (3,5), (6,5), (12,4), (13,2), (9,6), (4,12), (5,0). Paths followed by this set of requests are:

[1,3] = 1→MUX(1) →A→K→H'→DEMUX(15) →3

[2,4] = 2→MUX(4) →C→C'→DEMUX(4) →4

[3,5] = 3→MUX(5) →C→L→E'→DEMUX(9) →5

[6,5] = 6→MUX(10) →F→I→C'→DEMUX(5) →5

[12,4] = 12→MUX(4)→C→A→A'→DEMUX(0)→4

[13,2] = 13→MUX(7) →D→D'→DEMUX(6) →2

[9,6] = 9→MUX(9)->E->G->J->B'->DEMUX(2)->6

[4,12] = 4→MUX(12) →G→G'→DEMUX(12) →12

[5,0] = 5→MUX(15)→H→F→I→C'→DEMUX(4) →0

Number of requests matured successfully: 9

Average Path Length
= (3+2+3+3+3+2+4+2+4) /9 =2.888888

CASE III: Total number of requests: 11, Number of faults: 0

Let the 11 source destination pairs are (0,1), (4,5), (6,2), (7,8), (9,10), (12,14), (3,5), (1,11), (2,5), (15,7), (14,9). Paths followed by this set of requests are:

[0,1] = 0→MUX(0) →A→A'→DEMUX(1) →1

[4,5] = 4→MUX(4) →C→L→E'→DEMUX(9) →5

[6,2] = 6→MUX(6) →D→D'→DEMUX(6) →2

[7,8] = 7→MUX(15) →H→H'→DEMUX(15) →8

[9,10] = 9→MUX(9) →E→E'→DEMUX(9) →10

[12,14] = 12→MUX(12) →G→J→B'→DEMUX(2) →14

[3,5] = 3→MUX(3)→B→K→G'→DEMUX(13) →5

[1,11] = 1→MUX(11) →F→F'→DEMUX(11) →11

[2,5] = 2→MUX(10) →F→I→C'→DEMUX(5) →5

[15,7] = 15→MUX(7) →D→B→K→H'→DEMUX(15) →7

[14,9] = 14→MUX(8) →E→G→J→A'→DEMUX(1) →9

Number of requests matured successfully: 11
Average Path Length
= (2+3+2+2+2+3+3+2+3+4+4)/11 =2.727272

CASE IV: Total number of requests: 8, Number of faults: 0

Let the 8 source destination pairs are (1,3), (2,4), (3,5), (6,5), (9,6), (4,12), (5,14), (7,9). Paths followed by this set of requests are:

[1,3] = 1→MUX(1) →A→K→H'→DEMUX(15) →3

[2,4] = 2→MUX(4) →C→C'→ DEMUX(4) →4

[3,5] = 3→MUX(5) →C→L→E'→DEMUX(9) →5

[6,5] = 6→MUX(6) →D→B→K→G'→DEMUX(13) →5

[9,6] = 9→MUX(9) →E→I→D'→DEMUX(6) →6

[4,12] = 4→MUX(14) →H→J→A'→DEMUX(0) →12

[5,14] = 5→MUX(15) →F→F'→ DEMUX(14) →14

[7,9] = 7→MUX(9) →E→E'→DEMUX(9) →9

Number of requests matured successfully: 8
Average Path Length
=(3+2+3+4+3+3+2+2) /8 =2.750000

B. Permutation Passability in presence of Faults

This section describes the permutation passibility of two networks ASEN-2 and IASEN in presence of faults. For this analysis node A is considered faulty.

i. ASEN-2 Network

CASE I: Total number of requests: 6, Number of faults: 1

Let the six source destination pairs are (4,2), (1,3), (9,6), (3,10), (8,2), (7,10). Paths followed by this set of requests are:

[4,2] = 4→MUX(4)→C→C1→A'→DEMUX(1) →2

[1,3] = 1→MUX(9)→E→G→C1→A'→DEMUX(1)→3

[9,6] = 9→MUX(9) →CLASH

[3,10] =3→MUX(3) →B→F1→F'→DEMUX(11) →10

[8,2] = 8→MUX(8) →E→A1→A'→DEMUX(1) →2

[7,10] =7→MUX(15)→H→H1→G1→E'→DEMUX(9) →10

Number of requests matured successfully: 5
Average Path Length (3+4+0+3+3+4)/5 =3.400

CASE II: Total number of requests: 9, Number of faults: 1

Let the nine source destination pairs are (1,3), (2,4), (3,5), (6,5), (12,4), (13,2), (9,6), (4,12), (5,0). Paths followed by this set of requests are:

[1,3] = 1→MUX(9) →E→A1→A'→DEMUX(1) →3

[2,4] = 2→MUX(2)→B→B1→D'→DEMUX(6)→4

[3,5] =3→MUX(11)→F→B1→A1→C'→DEMUX(4) →5

[6,5] = 6→MUX(6) →D→D1→D'→DEMUX(6) →5

[12,4] =12→MUX(4)→C→C1→C'→DEMUX(4) →4

[13,2] = 13→MUX(13) →CLASH

[9,6] = 9→MUX(9) →CLASH

[4,12] = 4→MUX(12)→G→G1→G'→DEMUX(12)→12

[5,0] = 5→MUX(5) →CLASH

Number of requests matured successfully: 6
Average Path Length:
=(3+3+4+3+3+0+0+3+0) / 6 =3.1666

CASE III: Total number of requests: 11, Number of faults: 1

Let the 11 source destination pairs are (0,1), (4,5), (6,2), (7,8), (9,10), (12,14), (3,5), (1,11), (2,5), (15,7), (14,9). Paths followed by this set of requests are:

[0,1] = 0→MUX(8)→E→A1→A'→DEMUX(0)→1

[4,5] = 4→MUX(4)→C→C1→C'→DEMUX(4) →5

[6,2] = 6→MUX(6)→D→D1→B'→DEMUX(3) →2

[7,8] = 7→MUX(15) →H→H1→F'→DEMUX(10) →8

[9,10] = 9→MUX(9) →E→E1→E'→DEMUX(8) →10

[12,14] = 12→MUX(12) →G→G1→G'→DEMUX(13) →14

[3,5] = 3→MUX(3) →B→B1→D'→DEMUX(6)->5

[1,11] = 1→MUX(1) →CLASH

[2,5] = 2→MUX(2)→B→D→D1→D'→DEMUX(6) →5

[15,7] = 15→CLASH

[14,9] =14→MUX(14)→H→F→F1→F'→DEMUX (10) →9

Number of requests matured successfully: 9
Average Path Length:
= (3+3+3+3+3+3+3+0+4+0+4) / 9 =3.2222

CASE IV: Total number of requests: 8, Number of faults: 1

Let the 8 source destination pairs are (1,3), (2,4), (3,5), (6,5), (9,6), (4,12), (5,14), (7,9). Paths followed by this set of requests are:

[1,3] = 1→MUX(9)→E→A1→A'→DEMUX(1) →3
[2,4] = 2→MUX(2)→B→B1→D'→DEMUX(6)→4
[3,5] = 3→MUX(3)→B→D→D1→D'→DEMUX(6) →5
[6,5] = 6→MUX(14) →CLASH
[9,6] = 9→MUX(9) →CLASH
[4,12] = 4→MUX(12) →G→G1→G'→DEMUX(12) →12
[5,14] = 5→MUX(5)→C→F1→H'→DEMUX(15) →14
[7,9] = 7→MUX(15)→H→H1→F'→DEMUX(10) →9

Number of requests matured successfully: 6
Average Path Length
= (3+3+4+0+0+3+3+3) / 6) = 3.1666

ii. IASEN Network

CASE I: Total number of requests: 6, Number of faults: 1

Let the six source destination pairs are (4,2), (1,3), (9,6), (3,10), (8,2), (7,10). Paths followed by this set of requests are:

[4,2] = 4→MUX(4)→C→L → F' →DEMUX(10) →2
[1,3] = 1 → MUX(3) → B → B' → DEMUX(3) →3
[9,6] = 9→MUX(9) →E→I →D' →DEMUX(6) →6
[3,10] = 3→MUX(11) →F→F' →DEMUX(10) →10
[8,2] = 8→MUX(8) →E→G→J→B' →MUX(2) →2
[7,10] = 7→MUX(7) →D→D'→DEMUX(6) →10

Number of requests matured successfully: 6
Average Path Length:
(3+2+3+2+4+2) / 6 = 2.666666

CASE II: Total number of requests: 9, Number of faults: 1

Let the nine source destination pairs are (1,3), (2,4), (3,5), (6,5), (12,4), (13,2), (9,6), (4,12), (5,0). Paths followed by this set of requests are:

[1,3] = 1→MUX(3) →B→B'→DEMUX(3) →3
[2,4] = 2→MUX(4) →C→L→E'→DEMUX(8) →4
[3,5] = 3→MUX(11) →F→I→C'→DEMUX(5) →5
[6,5] = 6→MUX(8) →E→E'→DEMUX(9) →5
[12,4] = 12→MUX(12) →G→G'→DEMUX(12) →4
[13,2] = 13→MUX(7) →D→D'→DEMUX(6) →2
[9,6] = 9→MUX(9) →E→G→J→B'→DEMUX(2) →6
[4,12] = 4→MUX(14) →H→J→A'→DEMUX(0) →12
[5,0] =5→MUX(15)→H→F→I→C'→DEMUX(4)→0

Number of requests matured successfully: 9
Average Path Length:
(2+3+3+2+2+2+4+3+4) /9 = 2.777777

CASE III: Total number of requests: 11, Number of faults: 1

Let the 11 source destination pairs are (0,1), (4,5), (6,2), (7,8), (9,10), (12,14), (3,5), (1,11), (2,5), (15,7), (14,9). Paths followed by this set of requests are:

[0,1] = 0→MUX(8) →E→E'→DEMUX(9) →1
[4,5] = 4 →MUX(4) →C→L→E'→DEMUX(9) →5
[6,2] = 6→MUX(6) →D→D'→DEMUX(6) →2
[7,8] = 7→MUX(15) →H→H'→DEMUX(15) →8
[9,10] = 9→MUX(11) →F→F'→DEMUX(10) →10
[12,14]= 12→MUX(12) →G→J→B'→DEMUX(2) →14
[3,5] = 3→MUX(3) →B →K → G' →DEMUX(13) →5
[1,11] = 1→MUX(9) →E →I →D'→DEMUX(7) →11
[2,5] = 2→MUX(10) →F→H→J→A'→DEMUX(1) →5
[15,7] = 15→MUX(7) →D→B→K→H'→DEMUX(15) →7
[14,9] = 4→MUX(14)→H→F→I→C'→DEMUX(1)→9

Number of requests matured successfully: 11
Average Path Length:
(2+3+2+2+2+3+3+2+4+4+4) /11 = 2.818181

CASE IV: Total number of requests: 8, Number of faults: 1

Let the eight source destination pairs are (1,3), (2,4), (3,5), (6,5), (9,6), (4,12), (5,14), (7,9). Paths followed by this set of requests are:

[1,3] = 1→MUX(3) →B→B'→DEMUX(3) →3
[2,4] = 2→MUX(4) →C→C'→ DEMUX(4) →4
[3,5] = 3→MUX(5) →C→L→E'→DEMUX(9) →5
[6,5] = 6→MUX(6)→D→B→K→G'→DEMUX(13)→5
[9,6] = 9→MUX(9) →E→I→D'→DEMUX(6) →6
[4,12] = 4→MUX(14) → H→J→A'→DEMUX(0) →12
[5,14] = 5→MUX(15) →F→F'→ DEMUX(14) →14

[7,9] = 7→MUX(9) →E→E'→DEMUX(9) →9

Number of requests matured successfully: 8
Average Path Length:
(2+2+3+4+3+3+2+2)/8 = 2.625000

C. Permutation Comparison of ASEN-2 and IASEN

This section compares the permutation passibility bahavior of ASEN-2 and IASEN. TABLE I shows the comparative data for permutation analysis in terms of requests.

TABLE I
COMPARATIVE PERMUTATION OF ASEN-2 AND IASEN

Criteria	**ASEN-2**	**IASEN**
Total number of requests	34	34
Requests matured in absence of faults	30	34
Requests matured in presence of faults	26	34
Total path-length in absence of faults	98	94
Total path-length in presence of faults	84	93
Average Path-length in absence of faults	3.26	2.76
Average Path-length in presence of faults	3.23	2.73

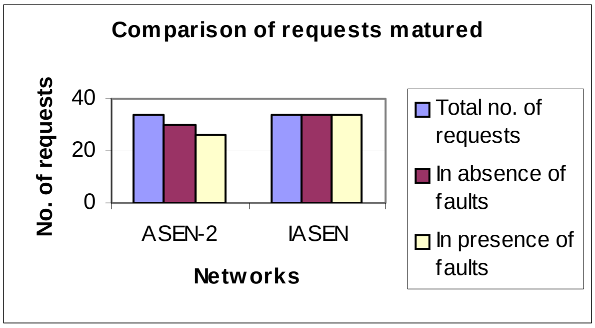

Fig. 6 Comparison of requests matured

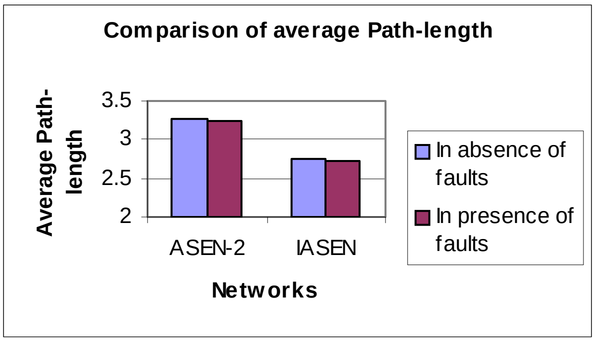

Fig. 7 Comparison of average Path-length

From Fig. 6 it is clear that IASEN handles all the requets in the presence and absence of faults. Whereas the performance of ASEN-2 degrades in the presence of faults. The path length used by various requests in IASEN is also lesser than the path length in ASEN-2 as shown in Fig. 7.

V. COST EFFECTIVENESS

To estimate the cost of a network the assumption is that the cost of a switch is proportional to the number of cross-points within a switch [1]. For example a 4x4 switch has 16 units of hardware cost whereas a 2x2 switch has 4 units. The cost functions for the various MINs are given in the Table II and the corresponding data values for cost are shown in TABLE III.

TABLE II
COST FUNCTIONS

Network	**Cost Function**
ASEN	$3N(1.5 \log_2 N - 1)$
IASEN	$3N(1.5 \log_2 N) - 20$

TABLE III
COST VALUES FOR DIFFERENT NETWORK SIZE

Network Size (Log N)	**Cost**	
	ASEN-2	**IASEN**
4	7.90	8.06
5	9.17	9.35
6	10.58	10.74
7	11.75	11.89
8	13.04	13.16
9	14.17	14.28
10	15.39	15.49

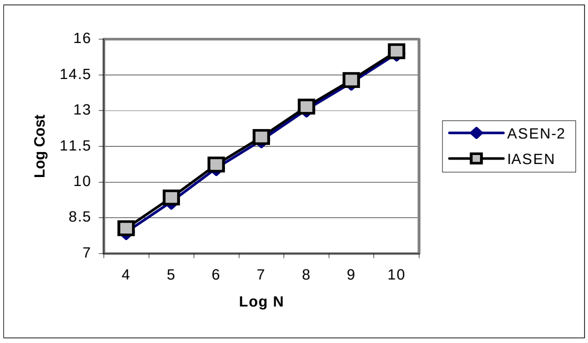

Fig. 9 Cost Comparison of ASEN-2 and IASEN

From Fig. 9 it is evident that ASEN-2 are IASEN are having same cost.

V. CONCLUSIONS

An Irregular Augmented Shuffle Exchange Network (IASEN) is designed using regular Augmented Shuffle Exchange Network (ASEN-2). IASEN is a dynamically re-routable and provides multiple paths of varying lengths between a given source-destination pair. It has been found that in an ASEN-2, six possible paths exist between any source-destination pair, whereas IASEN provides eight such paths. Therefore, the IASEN is much more fault-tolerant than ASEN-2. Permutation possibility behavior of IASEN in the presence and absence of faults also shows performance gain in comparison to ASEN-2.

REFERENCES

[1] Aggarwal H., and Bansal P.K., Routing and Path Length Algorithm for cost effective Modified Four Tree Network, IEEE TENCON, 2002, pp. 293-294.

[2] Aggarwal R., Aggarwal H. and Kaur L., “On Bandwidth analysis of Irregular Fault-tolerant Multistage Interconnection networks”, International Review on Computers and Software, Vol. 3, No. 2, March 2008, pp. 199-202.

[3] Aggarwal R. and Kaur L., “On Reliability Analysis of Fault-tolerant Multistage Interconnection Networks**”,** International Journal of Computer Science and Security**,** Vol. 2, No. 4, August 2008, pp. 1-8.

[4] Aggarwal R., Kaur L. and Aggarwal H., “Reliability Analysis of Fault-tolerant Irregular Multistage Interconnection Networks**”,** International Journal of Applied Engineering Research**,** Vol. 4, No. 6, June 2008, pp. 955-962.

[5] Bansal P.K., Joshi R.C. and Singh K., “On a fault-tolerant Multi-stage interconnection network”, International journal of Electronics and Electrical Engineering, Vol. 20, No.4, 1994, pp. 335-345.

[6] Bansal P.K., Joshi R.C, Singh K. and Siroha G.P., “Fault-tolerant Augmented Baseline Multistage Interconnection Network”, Proc. International Conference IEEE TENCON 91, INDIA, Aug. 1991, pp. 200-204.

[7] Bhuyan Laxmi N., Yang Qing and Agrawal Dharma P., “Performance of Multiprocessor Interconnection Networks”, IEEE Computer, Vol. 22, Feb. 1993, pp. 25-37.

[8] Duato Jose, Yalamanchili Sudhakar and Ni Lionel, “Interconnection Networks: An Engineering Approach”, IEEE Computer Society, 1997.

[9] Lubazewski M. and Coutois B., ”A Reliable Fail-safe System”, Parallel and Distributed Systems, IEEE Computer Society, Vol. 47, No. 2, Feb. 1998, pp. 236-241.

[10] Nitin, “On Analytic Bounds of Regular and Irregular Fault-tolerant Multistage Interconnection Networks”, International Conference on PDPTA, June 26-29, 2006, Vol. 1.

[11] Sadawarti Harsh and Bansal P.K., “Fault tolerant Irregular Augmented Shuffle Network” WSEAS International Conference on Computer Engineering and Applications, Gold Coast, Australia, 17-19, Jan, 2007.

[12] Sengupta J. and Bansal P.K, “Reliability and performance measures of regular and irregular multi-stage interconnection networks”, International Conference IEEE TENCON, 2000, pp. 531-536.

[13] Sengupta J. and Bansal P.K., “Performance Analysis of Regular and Irregular Dynamic MINs”, International Conference IEEE TENCON 99, Sept.1999, Cheju Island, Korea, pp. 427-430.

[14] Sharma Sandeep, Kalhon K.S., Bansal P.K. and Singh Kawaljeet, “Improved Irregular Augmented Shuffle Exchange Multistage Interconnection Network”, International Journal of Computer Science and Security, Vol. 2, No. 3, 2008, pp. 28-33.

[15] Sharma Sandeep, Kalhon K.S., Bansal P.K. and Singh Kawaljeet, “Irregular class of Multistage Interconnection Networks in Parallel Processing”, Journal of Computer Science, Vol. 4, No. 3, 2008, pp. 220-224.

[16] Sharma Sandeep, Kalhon K.S. and Bansal P.K., “On a class of Multistage Interconnection Networks in parallel processing” IJCSNS International Journal of Computer Science and Network Security, Vol.8, No.5, May 2008, pp. 287-291.

[17] Sheta A., Parameter Estimation of Software Reliability Growth Models by Particle Swarm Optimization, ICGST-AIML Journal, Vol. 7, No.7, June, 2007, pp.55-61.