

ON EFFICIENCY OF MULTI-STAGE INTERCONNECTION NETWORKS

Thesis submitted in fulfillment for the requirement of the degree of

DOCTOR OF PHILOSOPHY

by

Ved Prakash Bhardwaj



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING AND INFORMATION
TECHNOLOGY

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY WAKNAGHAT
SOLAN (H.P.)

DECEMBER 2014

ON EFFICIENCY OF MULTI-STAGE INTERCONNECTION NETWORKS

Thesis submitted in fulfillment for the requirement of the degree of

DOCTOR OF PHILOSOPHY

by

Ved Prakash Bhardwaj



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING AND INFORMATION
TECHNOLOGY

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY WAKNAGHAT
SOLAN (H.P.)

DECEMBER 2014

@ Copyright JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT
DECEMBER 2014
ALL RIGHTS RESERVED

For my loving Family and God

*Special Thanks to
Jaypee University of Information Technology*

ABSTRACT

Interconnection networks (INs) are the basic building block of all parallel processing systems. Multi-stage interconnection networks (MINs) are widely used in many parallel processing applications since it provides excellent performance at minimum cost with high reliability. It consists of more than one stage of small interconnection elements called switching elements (SEs) and links interconnecting them. MINs are the important class of INs. In this thesis, the author has focused on the efficiency of MINs and done his research in this direction in order to get the highly efficient MIN.

Basically, efficiency is the level of performance of a system. In this research work, the author has considered all the factors of performance that can measure the efficiency of a MIN e.g., bandwidth, the probability of acceptance, throughput, processor utilization, and processing power. Additionally, the efficiency depends on the condition of a MIN. It can be faulty or non-faulty. Faulty situations create the problem for data packets which are to be transmitted from the given source to the given destinations. Here the term “faulty situations” refers to the faulty SEs. This problem disturbs the data transmission process and creates a negative impact on the efficiency of the MIN.

To avoid this problem the solution is to design a MIN which possesses excellent fault tolerability with good performance. Literature survey shows that various MINs have been proposed to overcome this problem, e.g. modified alpha network (MALN), irregular augmented shuffle exchange network (IASEN). All the previously proposed MINs have less number of alternate paths between the source and destinations; have high cost and poor performances.

Towards the faulty switch problem, the author has proposed five new MINs named as advance irregular alpha multi-stage interconnection network

(AIAMIN), advance irregular augmented shuffle exchange network (AIASEN), irregular augmented shuffle exchange network-2 (IASEN-2), irregular advance omega network (IAON), and irregular augmented shuffle exchange network-3 (IASEN-3). All the proposed MINs have better fault sustainability than the earlier proposed MINs. Further, the MIN can be single switch fault tolerant or double switch fault tolerant. Single switch fault tolerant means the network can tolerate one faulty SE in each stage simultaneously and double switch fault tolerant means the network can tolerate two faulty SE in each stage simultaneously.

Further, the author has compared the performance of IAON with the MALN, and IASEN-2 in single switch faulty and non-faulty conditions. It is observed that IAON performs well in both conditions. Single switch faulty means the network has a single faulty switching element (SE) in each stage simultaneously. All the proposed MINs are single switch fault tolerant except the IAON. The specialty of IAON is that it is a double switch fault sustainable network. Additionally, the author has compared the performances of IAON and IASEN-3 in non-faulty and single switch faulty conditions and observed that IASEN-3 is better than the IAON on all the factors of performance. The advantage of IAON over IASEN-3 is that it is double switch fault tolerant network.

Further, the efficiency of a MIN is also affected by the crosstalk problem. In crosstalk, the electric and magnetic field of a signal creates disturbance in the network, which affects the other signal in a network e.g. telephone network. Crosstalk may happen in the hearing part of a voice conversation. In this way, it degrades the performance of a network. Due to the crosstalk, the data packets reached in the distorted form at the output side. To reduce the crosstalk, various approaches have been proposed in literature, e.g. space domain approach (SDA), time domain approach (TDA). Although, SDA reduces the crosstalk however it is costly and time consuming approach and therefore, it does not have any practical use in this direction. According to TDA, only a single communication signal should be passed through a SE

simultaneously. This approach is less costly and less time consuming than the space domain approach.

Further, there are various methods have been proposed which are based on TDA e.g. window method, improved window method. These methods are basically applicable to omega network and do the scheduling of messages in such a way that the conflicting messages are transmitted in the different passes. A big number of passes are required in case of window and improved window method to get the crosstalk free routes. Based on the TDA, a new network called the crosstalk free modified omega network (CFMON) is proposed by Sonia Kaur.

It provides crosstalk free routes in maximum four passes. Towards the crosstalk problem, the author has proposed address selection algorithm (ASA) and destination based modified omega network (DBMON) with its scheduling algorithm named as destination based scheduling algorithm (DBSA). ASA is applicable on omega network and clos network; however, it does not perform well for the network of larger sizes. DBSA is applicable on destination based modified omega network (DBMON). It takes only two passes to make the network crosstalk free. DBMON takes less transmission time than the CFMON and more efficient than the CFMON.

Hence, in this dissertation the research work of the author focuses on increasing efficiency of MIN. Therefore, he has given his major contribution to the faulty switch problem and a small contribution to crosstalk problem.

ACKNOWLEDGEMENTS

It gives me immense pleasure to thank all those who have helped and encouraged me in the completion of my research work leading to my Doctorate degree in computer Science.

I would like to express my heart-felt gratitude to my supervisor, **Associate Prof. (Dr.) Nitin**, faculty, Department of Computer Science & Engineering and Information Technology, for his valuable guidance, support and encouragement throughout my research work. I am indebted to him for his valuable advice, constructive criticism and never ending patience at each step of my research work.

I owe my thanks to **Prof. (Dr.) S. P. Ghreera, Brig. (Retd.)** Head Department of Computer Science & Engineering and Information Technology, **Dr. Yashwant Singh, Dr. Vivek Sehgal, Prof. (Dr.) Sunil Bhooshan**, Head Department of Electronics and other DPMC (Doctoral Program Monitoring Committee) members for providing me assistance, moral support, valuable suggestions and necessary facilities during the course of my research work.

I also wish my gratitude to **Chief Operating Officer - Jaypee Education System Dr. Y. Medury, Director Brig. (Retd.) Balbir Singh** and **Dean (Academic and Research) Prof. Dr. T. S. Lamba** for their interest, encouragement and help during the course of my research tenure at JUIT. I would like to thank the authorities of Jaypee University of Information Technology, Waknaghat for providing the financial support during my research work.

I also like to thank Mr. Shriram, Lab. Technicians of all computer labs in JUIT, for their help and cooperation.

Thanks are due for my dear friends Rashmi Sharma, Piyush Chauhan, Sharad and Manoj Gaur for their support and encouragement during my research work.

Words cannot express my humble gratitude and abyss regards to my dear parents, younger brother Abhimanyu, sister Amita and other family members for their affectionate encouragement and blessings to complete this research work.

I thank to GOD, especially Lord Hanuman (Ji) and Shri Ram (Ji) for showering their blessings always to provide me enormous patience, inspiration, faith and strength to carry out this research work.

(Ved Prakash Bhardwaj)



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY

(Established by H.P. State Legislative vide Act No. 14 of 2002)
P.O. DumeharBani, Kandaghat, Distt. Solan – 173234 (H.P.) INDIA

Website : www.juit.ac.in

Phone No. (91) 07192-257999 (30 Lines)

Fax: (91) 01792 245362

Date: 12, December, 2014

DECLARATION

I hereby declare that the work reported in the Ph.D. thesis entitled **“On Efficiency of Multi-Stage Interconnection Networks”** submitted at **Jaypee University of Information Technology, Wagnaghat India**, is an authentic record of my work carried out under the supervision of Associate Professor Dr. Nitin. I have not submitted this work elsewhere for any other degree or diploma.

(Signature of the Scholar)

Ved Prakash Bhardwaj

Department Of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Wagnaghat, India

Date (12, December, 2014)



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY

(Established by H.P. State Legislative vide Act No. 14 of 2002)
Waknaghat, P.O. Dumehar Bani, Kandaghat, Distt. Solan – 173234 (H.P.) INDIA

Website : www.juit.ac.in

Phone No. (91) 07192-257999 (30 Lines)

Fax: (91) 01792 245362

Date: 12, December, 2014

CERTIFICATE

This is to certify that the work reported in the Ph.D. thesis entitled “**On Efficiency of Multi-Stage Interconnection Networks**”, submitted by **Ved Prakash Bhardwaj** at **Jaypee University of Information Technology, Waknaghat, India**, is a bonafide record of his original work carried out under my supervision. This work has not been submitted elsewhere for any other degree or diploma.

(Signature of Supervisor)

Dr. Nitin

Associate Professor

Department Of Computer Science & Engineering and Information Technology

Date (12, December, 2014)

LIST OF FIGURES

Figure Number	Caption	Page Number
Figure 2.1	8×8 Omega Network	8
Figure 2.2	2-Stage Interconnection Network	11
Figure 2.3	16×16 IASEN	13
Figure 2.4	16×16 MALN	14
Figure 2.5	16×16 IMABN	15
Figure 2.6	16×16 IMABN-2	16
Figure 2.7	16×16 AON	17
Figure 2.8	Crosstalk in an electro optic SE	18
Figure 2.9	Legal passes to avoid the Crosstalk	19
Figure 3.1	16×16 AIASEN	22
Figure 3.2	Sending request from 0 to 3	25
Figure 3.3	Sending request from 0 to 3 through intermediate SEs	27
Figure 3.4	Sending request from 0 to 3 through intermediate SEs	28
Figure 3.5	Sending request from 2 to 1 through intermediate SEs	29
Figure 3.6	16×16 IASEN-2	31
Figure 3.7	16×16 AIAMIN	38
Figure 3.8	Sending request from SE A to SE I	41
Figure 3.9	Sending request from SE I to SE L	41
Figure 3.10	Sending request from SE L to SE O	42
Figure 3.11	Sending request from source 0 to SE E	43

Figure 3.12	Sending request from SE E to SE J	43
Figure 3.13	Sending request from SE J to SE M	43
Figure 3.14	Sending request from SE M to SE O	44
Figure 3.15	Sending request from source 0 to F	45
Figure 3.16	Sending request from SE F to SE K.	45
Figure 3.17	Sending request from K to M	46
Figure 3.18	Availability of paths between SEs B and T	47
Figure 3.19	AIAMIN with auxiliary links	48
Figure 3.20	Availability of extra alternate paths between SEs B and T	48
Figure 4.1	16×16 IAON	54
Figure 4.2	Redundancy Graph of IAON	55
Figure 4.3	Routing Algorithm of IAON	56
Figure 4.4	Primary Routes of Source 5 in IAON	58
Figure 4.5	First Alternate Routes of Source 5 in IAON	59
Figure 4.6	Second Alternate Routes of Source 5 in IAON	60
Figure 4.7	16×16 IASEN-3	63
Figure 4.8	Routing Algorithm of IASEN-3	64
Figure 4.9	Sample Network	70
Figure 4.10	Calculation of Data Transmission Time.	71
Figure 4.11	Comparison of BW for MALN, IASEN-2 and IAON when N=16	75
Figure 4.12	Comparison of BW for MALN, IASEN-2 and IAON when N=32	75
Figure 4.13	Comparison of BW for MALN, IASEN-2 and IAON when N=64	76

Figure 4.14	Comparison of BW for MALN, IASEN-2 and IAON when N=128	76
Figure 4.15	Comparison of PA for MALN, IASEN-2 and IAON	77
Figure 4.16	Comparison of TP for MALN, IASEN-2 and IAON when N=16 in Non-Faulty Condition	77
Figure 4.17	Comparison of TP for MALN, IASEN-2 and IAON when N=32 in Non-Faulty Condition	78
Figure 4.18	Comparison of TP for MALN, IASEN-2 and IAON when N=64 in Non-Faulty Condition	78
Figure 4.19	Comparison of TP for MALN, IASEN-2 and IAON when N=128 in Non-Faulty Condition	78
Figure 4.20	Comparison of TP for MALN, IASEN-2 and IAON when N=16 in Single Switch Faulty Condition	79
Figure 4.21	Comparison of TP for MALN, IASEN-2 and IAON when N=32 in Single Switch Faulty Condition	79
Figure 4.22	Comparison of TP for MALN, IASEN-2 and IAON when N=64 in Single Switch Faulty Condition	80
Figure 4.23	Comparison of TP for MALN, IASEN-2 and IAON when N=128 in Single Switch Faulty Condition	80
Figure 4.24	Comparison of PU for MALN, IASEN-2 and IAON when N=16 in Non-Faulty Condition	81
Figure 4.25	Comparison of PU for MALN, IASEN-2 and IAON when N=32 in Non-Faulty Condition	81
Figure 4.26	Comparison of PU for MALN, IASEN-2 and IAON when N=64 in Non-Faulty Condition	81
Figure 4.27	Comparison of PU for MALN, IASEN-2 and IAON when N=128 in Non-Faulty Condition	82
Figure 4.28	Comparison of PU for MALN, IASEN-2 and IAON when N=16 in Single Switch Faulty Condition	82
Figure 4.29	Comparison of PU for MALN, IASEN-2 and IAON when N=32 in Single Switch Faulty Condition	83
Figure 4.30	Comparison of PU for MALN, IASEN-2 and IAON when N=64 in Single Switch Faulty Condition	83

Figure 4.31	Comparison of PU for MALN, IASEN-2 and IAON when N=128 in Single Switch Faulty Condition	83
Figure 4.32	Comparison of PP for MALN, IASEN-2 and IAON when N=16 in Non-Faulty Condition	84
Figure 4.33	Comparison of PP for MALN, IASEN-2 and IAON when N=32 in Non-Faulty Condition	84
Figure 4.34	Comparison of PP for MALN, IASEN-2 and IAON when N=64 in Non-Faulty Condition	85
Figure 4.35	Comparison of PP for MALN, IASEN-2 and IAON when N=128 in Non-Faulty Condition	85
Figure 4.36	Comparison of PP for MALN, IASEN-2 and IAON when N=16 in Single Switch Faulty Condition	86
Figure 4.37	Comparison of PP for MALN, IASEN-2 and IAON when N=32 in Single Switch Faulty Condition	86
Figure 4.38	Comparison of PP for MALN, IASEN-2 and IAON when N=64 in Single Switch Faulty Condition	87
Figure 4.39	Comparison of PP for MALN, IASEN-2 and IAON when N=128 in Single Switch Faulty Condition	87
Figure 4.40	Comparison of BW for IASEN-2, IAON and IASEN-3 when N=16	89
Figure 4.41	Comparison of BW for IASEN-2, IAON and IASEN-3 when N=32	90
Figure 4.42	Comparison of BW for IASEN-2, IAON and IASEN-3 when N=64	90
Figure 4.43	Comparison of BW for IASEN-2, IAON and IASEN-3 when N=128	90
Figure 4.44	Comparison of PA for IASEN-2, IAON and IASEN-3	91
Figure 4.45	Comparison of TP for IASEN-2, IAON and IASEN-3 when N=16 in Non-Faulty Condition	92
Figure 4.46	Comparison of TP for IASEN-2, IAON and IASEN-3 when N=32 in Non-Faulty Condition	92
Figure 4.47	Comparison of TP for IASEN-2, IAON and IASEN-3 when N=64 in Non-Faulty Condition	92
Figure 4.48	Comparison of TP for IASEN-2, IAON and IASEN-3 when N=128 in Non-Faulty Condition	93

Figure 4.49	Comparison of TP for IASEN-2, IAON and IASEN-3 when N=16 in Single Switch Faulty Condition	93
Figure 4.50	Comparison of TP for IASEN-2, IAON and IASEN-3 when N=32 in Single Switch Faulty Condition	94
Figure 4.51	Comparison of TP for IASEN-2, IAON and IASEN-3 when N=64 in Single Switch Faulty Condition	94
Figure 4.52	Comparison of TP for IASEN-2, IAON and IASEN-3 when N=128 in Single Switch Faulty Condition	95
Figure 4.53	Comparison of PU for IASEN-2, IAON and IASEN-3 when N=16 in Non-Faulty Condition	95
Figure 4.54	Comparison of PU for IASEN-2, IAON and IASEN-3 when N=32 in Non-Faulty Condition	96
Figure 4.55	Comparison of PU for IASEN-2, IAON and IASEN-3 when N=64 in Non-Faulty Condition	96
Figure 4.56	Comparison of PU for IASEN-2, IAON and IASEN-3 when N=128 in Non-Faulty Condition	97
Figure 4.57	Comparison of PU for IASEN-2, IAON and IASEN-3 when N=16 in Single Switch Faulty Condition	97
Figure 4.58	Comparison of PU for IASEN-2, IAON and IASEN-3 when N=32 in Single Switch Faulty Condition	98
Figure 4.59	Comparison of PU for IASEN-2, IAON and IASEN-3 when N=64 in Single Switch Faulty Condition	98
Figure 4.60	Comparison of PU for IASEN-2, IAON and IASEN-3 when N=128 in Single Switch Faulty Condition	99
Figure 4.61	Comparison of PP for IASEN-2, IAON and IASEN-3 when N=16 in Non-Faulty Condition	99
Figure 4.62	Comparison of PP for IASEN-2, IAON and IASEN-3 when N=32 in Non-Faulty Condition	100
Figure 4.63	Comparison of PP for IASEN-2, IAON, and IASEN-3 when N=64 in Non-Faulty Condition	100
Figure 4.64	Comparison of PP for IASEN-2, IAON and IASEN-3 when N=128 in Non-Faulty Condition	101
Figure 4.65	Comparison of PP for IASEN-2, IAON and IASEN-3 when N=16 in Single Switch Faulty Condition	101

Figure 4.66	Comparison of PP for IASEN-2, IAON and IASEN-3 when N=32 in Single Switch Faulty Condition	102
Figure 4.67	Comparison of PP for IASEN-2, IAON and IASEN-3 when N=64 in Single Switch Faulty Condition	102
Figure 4.68	Comparison of PP for IASEN-2, IAON and IASEN-3 when N=64 in Single Switch Faulty Condition	103
Figure 4.69	Comparison of TP for IAON when N=16 in Non-Faulty, Single Switch Faulty, and Double Switch Faulty Conditions	105
Figure 4.70	Comparison of TP for IAON when N=32, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	105
Figure 4.71	Comparison of TP for IAON when N=64, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	106
Figure 4.72	Comparison of TP for IAON when N=128, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	106
Figure 4.73	Comparison of PU for IAON when N=16, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	107
Figure 4.74	Comparison of PU for IAON when N=32, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	107
Figure 4.75	Comparison of PU for IAON when N=64, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	108
Figure 4.76	Comparison of PU for IAON when N=128, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	108
Figure 4.77	Comparison of PP for IAON when N=16, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	109
Figure 4.78	Comparison of PP for IAON when N=32, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	109
Figure 4.79	Comparison of PP for IAON when N=64, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	110
Figure 4.80	Comparison of PP for IAON when N=128, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	110
Figure 5.1	8×8 CFMON	114

Figure 5.2	Address Selection Algorithm	115
Figure 5.3	Combination Matrix	116
Figure 5.4	Transpose of Matrix	116
Figure 5.5	Selection of Middle 4 Rows	117
Figure 5.6	Transmission of Selected Addresses	118
Figure 5.7	Transmission of SAs of High Magnitude	119
Figure 5.8	8×8 Clos Network	120
Figure 5.9	8×8 Clos Network with Crosstalk	121
Figure 5.10	Combination Matrix	122
Figure 5.11	Transpose Matrix	122
Figure 5.12	Selected Rows	122
Figure 5.13	First Pass of Selected SAs and their DAs	124
Figure 5.14	8×8 DBMON	125
Figure 5.15	Routing of Bit '0' in DBMON	126
Figure 5.16	Routing of Bit '1' in DBMON	126
Figure 5.17	DBSA	127
Figure 5.18	PIA	128
Figure 5.19	Crosstalk Free First Pass through DBMON	131
Figure 5.20	Comparison Based on Maximum NOP	132
Figure 5.21	Comparison Based on Maximum Transmission Time	133

LIST OF TABLES

Table Number	Caption	Page Number
Table 4.1	Symbol Table	52
Table 4.2	Symbol Table	53
Table 4.3	Load on Given Load Factors	66
Table 4.4	Average PA	88
Table 4.5	Average BW	88
Table 4.6	Average PA	103
Table 4.7	Average BW	104
Table 5.1	Symbol Table	113
Table 5.2	SAs and DAs	116
Table 5.3	Pairing of Rows	117
Table 5.4	Addition of Rows	117
Table 5.5	Subtraction of Rows	118
Table 5.6	SAs and DAs	121
Table 5.7	Pairing of Rows	122
Table 5.8	Addition of Corresponding Bits	123
Table 5.9	Subtraction	123
Table 5.10	SAs and DAs	129
Table 5.11	First Pass of CFMON	130
Table 5.12	Second Pass of CFMON	130
Table 5.13	First Pass of DBMON	131
Table 5.14	Second Pass of DBMON	132
Table 5.15	BW Comparison	134
Table 5.16	Load on given Load Factors	134

TABLE OF CONTENTS

	PAGE NO
INNER FIRST PAGE	i
ABSTRACT	v
ACKNOWLEDGEMENT	viii
DECLARATION BY SCHOLAR	ix
SUPERVISOR'S CERTIFICATE	x
LIST OF FIGURES	xi
LIST OF TABLES	xviii
TABLE OF CONTENTS	xix
CHAPTER 1	
INTRODUCTION	1-6
1.1 Problem Statement and Contributions	2
1.2 Outlines of thesis	6
CHAPTER 2	
PRELIMINARIES AND BACKGROUND	7-20
2.1 Multi-stage Interconnection Networks	8
2.2 Faulty Switches in Multi-stage Interconnection Networks	10
2.3 Earlier Proposed MINs	12
2.3.1 Irregular Augmented Shuffle Exchange Network	12
2.3.2 Modified Alpha Network (MALN)	13
2.3.3 Irregular Modified Augmented Baseline Network	14
2.3.4 Irregular Modified Augmented Baseline Network-2	16
2.3.5 Advance Omega Network	17
2.4 Crosstalk in Multi-stage Interconnection Networks	17
2.5 Methods to Reduce the Crosstalk in MINs	18
2.5.1 Space Domain Approach	18
2.5.2 Time Domain Approach	19
2.6 Conclusion	20
CHAPTER 3	
COMPARATIVE ANALYSIS OF FAULT TOLERANT MINs BASED ON COST AND AVAILABILITY OF PATHS	21-49
3.1 Introduction and Motivation	21
3.2 Related Work	21

3.3	AIASEN	22
3.3.1	Structure of AIASEN	22
3.3.2	Routing Algorithm of AIASEN	23
3.3.2.1	Proof of Correctness	24
3.3.3	Path Availability	29
3.3.4	Cost Analysis	30
3.4	IASEN-2	30
3.4.1	Structure of IASEN-2	31
3.4.2	Routing Algorithm of IASEN-2	32
3.4.2.1	Proof of Correctness	34
3.4.3	Cost Analysis	37
3.5	AIAMIN	37
3.5.1	Structure of AIAMIN	37
3.5.2	Routing Algorithm of AIAMIN	39
3.5.2.1	Proof of Correctness	40
3.5.2.2	Path Availability	46
3.5.3	Cost Analysis	48
3.6	Conclusion and Future Scope of Work	49

CHAPTER 4

COMPARATIVE ANALYSIS OF FAULT TOLERANT MINs BASED ON THE VARIOUS PERFORMANCE FACTORS 50-111

4.1	Introduction	50
4.2	Related Work	51
4.3	Irregular Advance Omega Network	54
4.3.1	Structure of Irregular Advance Omega Network	54
4.3.2	Routing Algorithm of IAON	55
4.3.2.1	Proof of Correctness	57
4.4	Irregular Augmented Shuffle Exchange Network-3	62
4.4.1	Structure of Irregular Augmented Shuffle Exchange Network-3	62
4.4.2	Routing Algorithm of IASEN-3	63
4.5	Performance Analysis and Simulation Results	65
4.5.1	Load Factor	66
4.5.2	Calculation of Bandwidth	67
4.5.3	Calculation of Probability of Acceptance	69
4.5.4	Calculation of Transmission Time	69
4.5.5	Calculation of Throughput	72
4.5.6	Calculation of Processor Utilization	73
4.5.7	Calculation of Processing Power	74
4.5.8	Bandwidth for MALN, IASEN-2 and IAON	75
4.5.9	Probability of Acceptance for MALN, IASEN-2 and IAON	76
4.5.10	Throughput for MALN, IASEN-2 and IAON in Non-Faulty Condition	77
4.5.11	Throughput for MALN, IASEN-2 and IAON in Single Switch Faulty Condition	79
4.5.12	Processor Utilization for MALN, IASEN-2 and IAON in Non-Faulty Condition	80
4.5.13	Processor Utilization for MALN, IASEN-2 and IAON	82

in Single Switch Faulty Condition	
4.5.14 Processing Power for MALN, IASEN-2 and IAON in Non-Faulty Condition	82
4.5.15 Processing Power for MALN, IASEN-2 and IAON in Single Switch Faulty Condition	85
4.5.16 Bandwidth for IASEN-2, IAON and IASEN-3	89
4.5.17 Probability of Acceptance for IASEN-2, IAON and IASEN-3	91
4.5.18 Throughput for IASEN-2, IAON and IASEN-3 in Non-Faulty Condition	91
4.5.19 Throughput for IASEN-2, IAON and IASEN-3 in Single Switch Faulty Condition	93
4.5.20 Processor Utilization for IASEN-2, IAON and IASEN-3 in Non-Faulty Condition	95
4.5.21 Processor Utilization for IASEN-2, IAON and IASEN-3 in Single Switch Faulty Condition	97
4.5.22 Processing Power for IASEN-2, IAON and IASEN-3 in Non-Faulty Condition	99
4.5.23 Processing Power for IASEN-2, IAON and IASEN-3 in Single Switch Faulty Condition	101
4.5.24 Throughput for IAON in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	104
4.5.25 Processor Utilization for IAON in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	106
4.5.26 Processing Power for IAON in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions	109
4.6 Conclusion and Future Scope of the Work	111
CHAPTER 5	
MINIMIZING CROSSTALK IN MINs	112-135
5.1 Introduction	112
5.2 Related Work	113
5.2.1 Crosstalk Free Modified Omega Network	113
5.3 Address Selection Algorithm	114
5.3.1 Applying ASA on Clos Network	120
5.4 Destination Based Modified Omega Network	125
5.4.1 Structure of DBMON	125
5.4.2 Internal Routing in DBMON	125
5.4.3 Destination Based Scheduling Algorithm	126
5.4.4 Path Information Algorithm	128
5.4.5 Performance Evaluation Parameters	128
5.4.5.1 Maximum Number of Passes	129
5.4.5.2 Transmission Time (t)	129
5.4.5.3 Maximum Transmission Time (tt)	129
5.4.6 Performance Comparison of CFMON and DBMON	129
5.4.6.1 Comparison Based on Maximum Number of Passes	132
5.4.6.2 Comparison Based on Maximum Transmission Time	132
5.5 Conclusion and Future Scope of the Work	135

CHAPTER 6	
CONCLUSION AND FUTURE WORK	136-137
REFERENCES	138-146

CHAPTER 1

INTRODUCTION

It is observed that various applications of real world are based on a parallel processing system (PPS) e.g. air traffic control, weather forecasting, robot vision. PPS is based on divide and conquer method [1,2]. It divides the task into multiple subtasks and executes them concurrently. It consists of following three units:

- Processing Elements
- Memory Modules
- Interconnection Network

Processing elements (PEs) do the computation task, memory modules store the given data, and the interconnection network (IN) performs the data transfer operation between the PEs and memory modules [3,4, 5]. In this way, INs work as an effective medium in PPS. It can be defined as:

“Interconnection Networks should be designed to transfer the maximum amount of information within the least amount of time (and cost, power constraints) so as not to bottleneck the system [6]”.

The INs exist for a small digital system that have at least two components to interconnect them to a large communication system that may have several processing nodes, memory units to do the parallel computation e.g. computer system[1, 2]. In a computer system, it connects processors to memories, I/O devices to I/O controllers.

IN is a major factor for high performance PPS as it has the capacity to provide high bandwidth and excellent communication between the connecting devices. The multi-stage interconnection network (MIN) is an important class

of interconnection network since it has the capacity of providing fast communication between the sources and the given destinations at the minimum cost [5,6] .

1.1 Problem Statement and Contributions

In this thesis, the author has focused his attention on multi-stage interconnection network. MIN is a cost effective IN and has many advantages over the other INs. Some of them are as follows [4,5,6]:

1. *It has Low latency.*
2. *The system is scalable.*
3. *It provides fault tolerance.*
4. *It provides cost effectiveness.*
5. *The probability of acceptance of data packets is high.*
6. *It posses high performance.*

Further, MINs have their own challenges and to increase the efficiency one has to reduce these challenges. Here the author has mentioned some of the challenges of a MIN which are as follows:

1. Faulty MIN
2. Crosstalk
3. Stable Matching
4. Network on Chip
5. Load Balancing

In this thesis, the author has mainly focused on the faulty MIN problem [7-11] and also gave some attention to crosstalk problem [12-14]. Faulty MIN means faulty switch problem in MINs.

Further, faulty switching elements (SEs) hamper the data transmission process. Due to these faulty switches, data packets do not get their given destination. Faulty condition of a network degrades its performance [15-21]. De-

signing a highly fault tolerant MIN is the best solution of this problem [16-22]. Towards the faulty switch problem, many INs have been proposed in recent years e.g. modified alpha network (MALN), irregular augmented shuffle exchange network (IASEN). All the proposed MINs have the following shortcomings:

1. Poor performance
2. Less number of alternate paths
3. Less fault tolerant
4. Posses high cost

To solve the faulty switch problem, the author has proposed five interconnection networks which are as follows:

1. Advance irregular alpha multi-stage interconnection network (AIAMIN)
2. Advance irregular augmented shuffle exchange network (AIASEN)
3. Irregular augmented shuffle exchange network-2 (IASEN-2)
4. Irregular advance omega network (IAON)
5. Irregular augmented shuffle exchange network-3 (IASEN-3)

AIAMIN [8] is a single switch fault tolerant network and it consists of four stages. Single switch fault tolerant means the network can tolerate a faulty switch in each stage simultaneously. The AIAMIN is compared with MALN and it is observed that AIAMIN has more alternate paths than the MALN and produce better fault sustainability than the MALN.

AIASEN[11] and IASEN-2 [10] consist of five and four stages respectively. Both INs are single switch fault tolerant networks and compared with irregular augmented shuffle exchange network (IASEN). It is observed that both the proposed MINs have better fault tolerance capacity than the IASEN. Further, IASEN has less number of alternate paths than the AIASEN and IASEN-2.

The design pattern of IAON [7] is inspired by advance omega network. IAON is a double switch fault tolerant network and it consists of three stages. Dou-

ble switch fault tolerant means the network can sustain two faulty switches in each stage simultaneously. It is compared with MALN and IASEN-2. IAON has better fault tolerability than the MALN and IASEN-2. Further, results show that IAON has better performance than the MALN and IASEN-2 in faulty and non-faulty conditions.

The structure of IASEN-3[9] is based on IASEN-2. IASEN-3 is a single switch fault tolerant network and it consists of three stages. IASEN-3 has better fault tolerability and better performance than the IAON. However, IAON is double switch fault tolerant network.

The contributions towards this research are published and are as follows:

- V. P. Bhardwaj and Nitin, Message broadcasting via a new fault tolerant irregular advanced omega network in faulty and non-faulty network environments, *Journal of Computer and Electrical Engineering*, Hindawi Publishing Corporation, vol. 2013, pp. 1-17, 2013.
- V. P. Bhardwaj and Nitin, A new fault tolerant routing algorithm for advance irregular alpha multi-stage interconnection network, *Advances in Intelligent and Soft Computing*, Springer-Verlag, ISSN: 1867-5662, pp. 979-987, 2012.
- V. P. Bhardwaj and Nitin, On performance analysis of IASEN-3 in faulty and non faulty network conditions, *AASRI Conference on Intelligent Systems and Control*, Elsevier, pp. 104-109, 2013.
- V. P. Bhardwaj and Nitin, A new fault tolerant routing algorithm for IASEN-2, *Proceedings of the 2nd IEEE International Conference on Advances in Computing and Communications*, pp. 199-202, 2012.
- V. P. Bhardwaj and Nitin, A new fault tolerant routing algorithm for advance irregular augmented shuffle exchange network, *Proceedings of the*

14th IEEE International Conference on Computer Modeling and Simulation, pp. 505-509, 2012.

Further, crosstalk [15, 23-27] is also an important factor that makes its negative impact on the efficiency of a MIN. It degrades the performance of the network and limits the size of the network. Literature review [15, 25-29] reveals that various approaches have been proposed earlier in this regard e.g. space domain approach (SPA), time domain approach (TDA). SPA is costly and time consuming than the TDA and therefore, the author has chosen the TDA to reduce the crosstalk problem. He has proposed address selection algorithm (ASA) [13, 14] in order to reduce the crosstalk in omega network. Further, the author has applied [13, 14] the ASA on clos network and got the crosstalk free routes in the minimum number of passes.

ASA does well for the small size network, but does not produce good results for large size networks. Therefore, the author has proposed a new interconnection network named as destination based modified omega network (DBMON) [12]. It is applicable from a network of small size to a network of large size. The DBMON[12] takes the least number of passes and provides crosstalk free routes than the crosstalk free modified omega network (CFMON) [28].

The contributions towards this research are published and are as follows:

- V. P. Bhardwaj and Nitin, On minimization of crosstalk conflicts in destination based modified omega network, *Journal of Information Processing Systems*, vol. 9, no. 2, pp. 301-314, 2013.
- V. P. Bhardwaj, Nitin, and Vipin Tyagi, An algorithmic approach to minimize the conflicts in an optical multi-stage interconnection network, *Communications in Computer and Information Science Series*, Springer-Verlag, vol. 191, ISSN: 1865-0929, pp. 568-576, 2011.

- V. P. Bhardwaj and Nitin, Applying address selection algorithm on nonblocking optical multi-stage interconnection network, Proceedings of the IEEE World Congress on Information and Communication Technologies, pp. 694-698, 2011.

1.2 Outline of the Thesis

This thesis has six chapters. In chapter 1, the author has given the introductory part of the thesis. Preliminaries and background is explained in chapter 2. Towards the faulty switch problem, single switch fault tolerant and double switch fault tolerant MINs are presented in chapter 3 and 4 respectively. Chapter 4 also shows the experimental results. Solutions against the crosstalk problem are explained in chapter 5. In chapter 6, the author has given conclusions of all chapters and also presents the future scope of the research work. References and list of publications are given after chapter 6.

CHAPTER 2

PRELIMINARIES AND BACKGROUND

The main task of interconnection network (IN) is to establish a fast and efficient communication among the processing nodes in a parallel computing system [1, 4, 5] . It can be classified in various categories based on the following strategies:

- Switching methodology
- Topology

According to the switching technique [3, 6, 15], it can be circuit switched network or packet switch network. In circuit switching, a physical path is established between a source and a destination. This technique is preferred for the messages of long size. The whole path is reserved for the message until it reaches at the given destination.

This is the major drawback of circuit switching technique [15]. In packet switching, the given message is divided into small packets for transmission from the sources to destinations through the network. In this technique, the path between the source and destination is not reserved for a single message and it is fully utilized by the data packets [6, 15].

Topology [1, 6, 15] is the pattern of the INs means it shows that how the switching elements (SEs) are connected to the other processing elements. Further, the connecting links in INs may be static or dynamic. In static, these links are fixed, e.g. ring, star, tree, mesh etc.

The static networks are also referred as the direct interconnection network since in this network each switch is directly connected to a processing element [15]. In dynamic, the links can be reconfigured e.g. bus, crossbar, mul-

ti-stage etc. The dynamic networks are also referred as the indirect interconnection network since some of the switches connected to the other switches [3, 4, 15].

2.1 Multi-stage Interconnection Networks

Multi-stage interconnection networks (MINs) are the most suitable network among all the previously proposed INs since it is dynamic, fault tolerant, cost effective and stellar in performance [1, 3, 5]. MINs come in the category of high speed networks. It has more than one stage and each stage consists of small interconnecting elements called the SEs.

Each stage in MIN connects with the previous and subsequent stage via the connecting links. At the one end it has the sources and at the other end it has the destinations. The sources and destinations are having connectivity via the given switching stages. Generally, the size of the SE is 2×2 in MINs. In figure 2.1, the author has shown the 8×8 omega network [16-19]. This network has 8 sources (S) and 8 destinations (D).

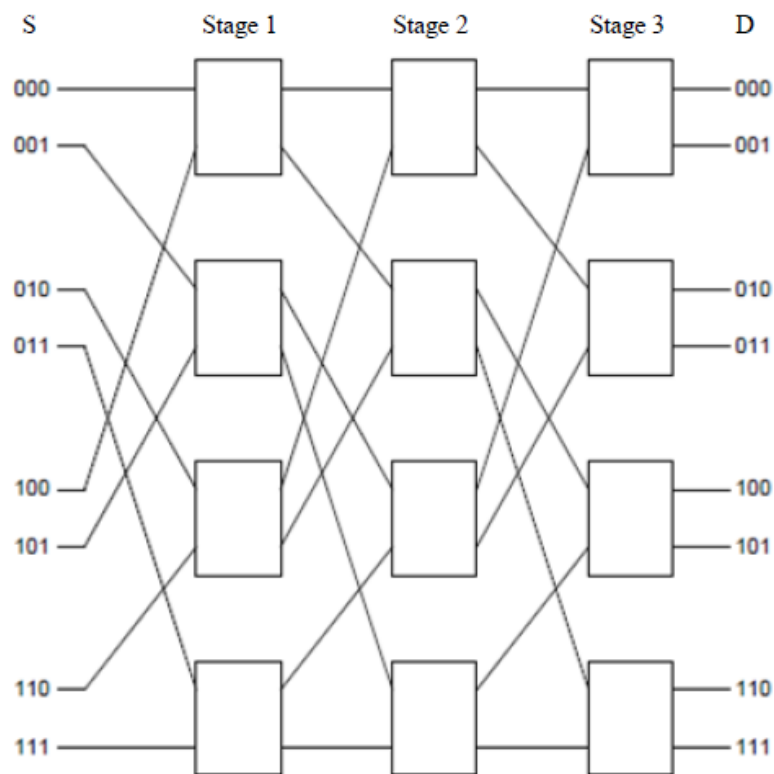


Figure 2.1. 8×8 Omega Network.

There are 3 stages in this network and each stage contains 4 SEs of size 2×2 for each one. In figure 2.1, the connectivity of all the given sources, stages, and destinations is shown. Further, MIN can be classified into the various categories based on the following parameters [1, 3, 15]:

- Number of paths
- Number of SEs
- Availability of paths

According to the number of paths [6, 15], MINs can be of single path nature or multipath nature. In single path, each source and destination are connected by a unique path in the network. This type of network cannot tolerate a single switch failure and therefore these are not used in the parallel communication system. The multipath networks provide more than one path between every pair of source and destination. High fault tolerability and good performance are the specialty of these types of networks.

According to SEs, MINs may be regular or irregular network [6,15]. In regular MIN, the number of SEs are same in each stage whereas in irregular MIN, the number of SEs are not same.

Blocking and nonblocking [15] are two major categories of MINs which depends on the availability of paths. In blocking MINs, SE of some stage may gets multiple requests from the source or SE of the previous stage. In this case, the path will be blocked and data packets will not move towards the destination end e.g. omega network.

In nonblocking MINs, each source has multiple alternate paths in order to route the data packets towards the destination end. However, no two or more than two sources have the same destination e.g. clos network.

Further, the efficiency of MINs is the major concern of this thesis [1, 4-6]. A MIN can be made efficient when it does not have any fault or that can transmit the data in crucial faulty situations. Therefore, the author has provided five fault tolerant MINs and compared the proposed MIN with the existing MINs.

Although efficiency of a MIN depends on the various factors, but this thesis mainly emphasis on the switch fault problem and gave some contribution towards the crosstalk problem. Crosstalk [22-29] is also a critical problem in MINs. The data packets get distortion because of crosstalk and therefore performance of a MIN is degraded.

2.2 Faulty Switches in Multi-stage Interconnection Networks

Faults in MINs halt the data transmission process [30-35]. All the performance factors like throughput, processor utilization, and processing power are also affected because of faulty situations [21, 22].

In this thesis, faulty situations refer to the faulty switch problem. Sending data packets from a source to the given destinations through MIN in critical fault situations are only being possible when the network is highly fault tolerant [36-42].

Further, transmission of data packets from a source to a given destination through the MIN is possible in following situations:

- When the network is non-faulty.
- When the processing elements including switches are not busy.
- In case if the network has faulty switching elements, then it must have the sufficient number of alternate paths.

In figure 2.2, a simple 2-stage interconnection network has been shown [7].

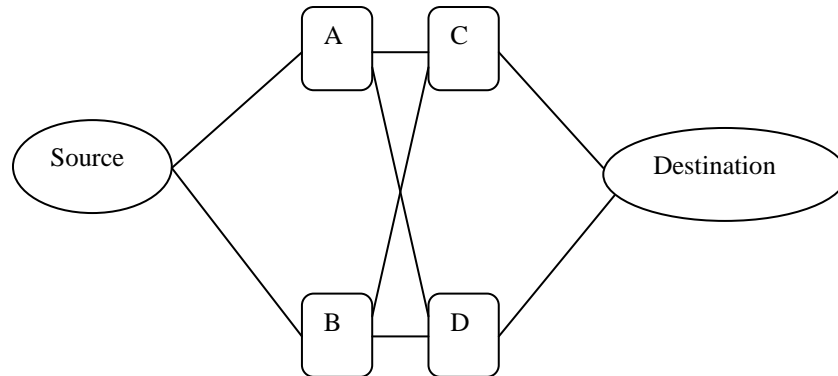


Figure 2.2. 2-Stage Interconnection Network.

In first stage it has 2 SEs i.e. A and B. In the second stage it has SEs C and D. Both SEs of the first stage are connected with SE C and D of second stage. To send the data from source to destination, the network provides the following alternate paths:

Path1: Source-A-C-Destination

Path2: Source-A-D-Destination

Path3: Source-B-C-Destination

Path4: Source-B-D-Destination

If the SE A is faulty then data packets can take any route via SE B. If both SEs of the first stage are faulty then data transmission will be stopped. In case, if required SE is busy then data packets have to wait for their chance. Therefore, it can be said that for excellent data transmission one has to design a MIN which has following qualities [43-49]:

- Provides excellent communication between source and destination
- Offers good bandwidth
- Posses high fault tolerability
- Takes minimum hardware cost

Based on the fault tolerability [50-54], the author has considered two types of MINs i.e. single switch fault tolerant and double switch fault tolerant. In single switch fault tolerant MIN; the MIN can sustain a single faulty SE in each stage simultaneously and transmit the data packets from the given source to the given destination effectively [55-58].

Further, the MIN that can sustain two faulty SE in each stage simultaneously known as double switch fault tolerant network [7]. In literature, various fault tolerant MINs have been proposed. However, obtaining a highly fault tolerant MIN is a critical challenge. To increase the fault tolerability, many INs have been explained in the next section.

2.3 Earlier Proposed MINs

2.3.1 Irregular Augmented Shuffle Exchange Network

This network [20] has m stages, where $m = \log_2 N / 2$ and N is the size of the network. In the first and last stage, it has an equal number of SEs i.e. 8 switches in the first stage of size 3×3 and 8 switches in the last stage of size 2×2 . The middle stage has four SEs of size 3×3 .

The source address, destination address, multiplexer, and demultiplexer are represented by S , D , Mux , and $Demux$ respectively. This network forms a loop in the first stage and provides multiple paths to avoid a fault. In IASEN, two SEs of each group are connected with each source and destination address with the help of Mux and $Demux$ respectively.

Figure 2.3 shows the structure of IASEN [20]. The IASEN is an irregular interconnection network with high cost and low accessibility. Routing of data packets in IASEN depends on condition of the network. If the network has faulty SEs then alternate path is used for data transmission. In each stage, the most suitable SE is selected for data transmission. The SE can be said suitable if it is free and non-faulty in order to receive a request.

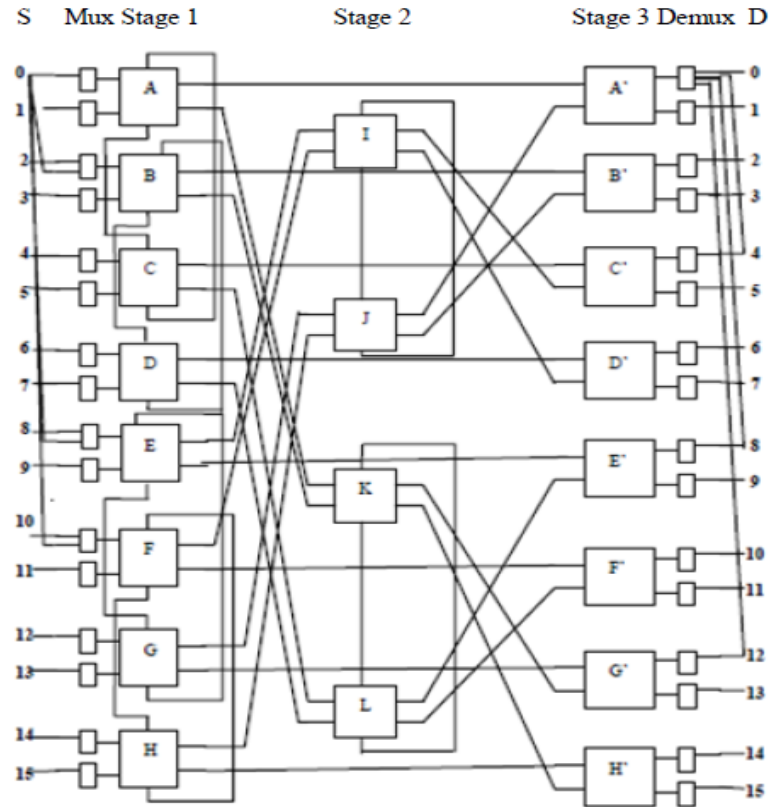


Figure 2.3. 16×16 IASEN.

2.3.2 Modified Alpha Network (MALN)

MALN [21] has N sources and N destinations with $n = \log_2 N$ stages. There are two subgroups in MALN. Each subgroup has $N/2$ sources and $N/2$ destinations. The network has connectivity with all the sources and destinations via multiplexers and demultiplexers. Connectivity of SEs through auxiliary links are exists in stage $n-3$, $n-2$, and $n-1$. In figure 2.4, the structure of MALN is shown. The source address, destination address, multiplexer, and demultiplexer are represented by S, D, Mux, and Demux respectively.

In MALN [21], routing of data packets is based on the binary values of source and destination addresses. MALN has two subnetworks i.e. G_0 and G_1 as shown in figure 2.4. Initially, the MSB of the destination address is checked in order to select a subnetwork. After selecting a subnetwork, data packets will be sent to the appropriate SE. Appropriate SE means it is free and non-faulty otherwise data packets are routed through auxiliary links to send them on another appropriate SE.

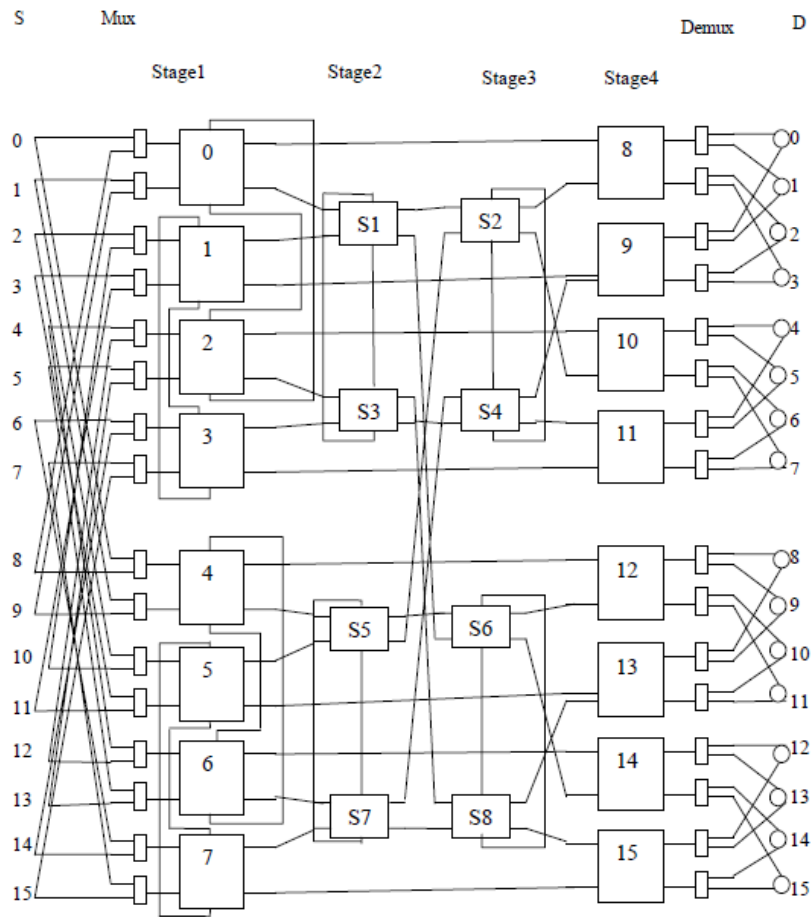


Figure 2.4. 16×16 MALN.

If the required SE cannot receive the data packets due to fault or any other problem, then the request will be dropped. In the next step, the same process is repeated for further data transmission. Data packets always send to non-faulty SE. If all the required SEs are faulty and then the request will be dropped otherwise send towards the given destination.

2.3.3 Irregular Modified Augmented Baseline Network

IMABN[22] stands for irregular modified augmented baseline network.

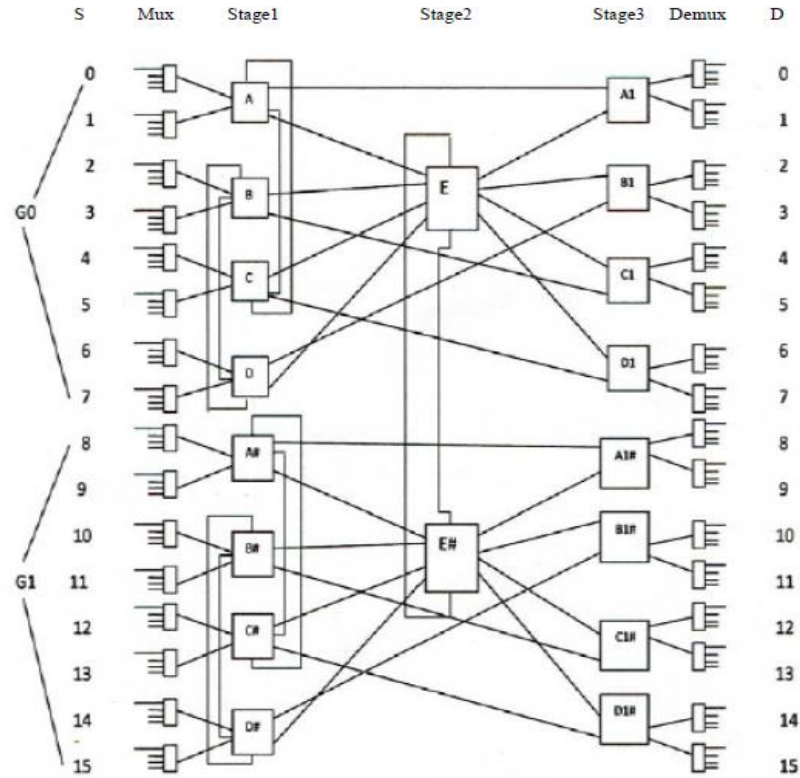


Figure 2.5. 16×16 IMABN.

A 16×16 IMABN [22] is shown in figure 2.5. The source address, destination address, multiplexer, and demultiplexer are represented by S, D, Mux, and Demux respectively. The given sources and destinations are further divided into two subgroups i.e. G_0 and G_1 . Every source is connected to both groups via multiplexers e.g. SE A, B, C, and D exist in stage 1 and comes in subgroup G_0 . These SEs make a conjugate subset. SE A and B make a conjugate pair, and SE A and C makes a conjugate pair.

Similarly, SE of subgroup G_1 has the conjugate pairs. In IMABN the size of each Mux and Demux are 4×1 and 1×4 respectively. The size of each SE in the first stage, second stage, and third stage is 3×3, 5×5, and 2×2 respectively. SEs of middle stage also have the auxiliary links. IMABN is based on the irregular topology. IMABN is a dynamic interconnection network and has more alternate path than the MABN [22].

2.3.4 Irregular Modified Augmented Baseline Network-2

Irregular modified augmented baseline network-2 (IMABN-2) is a five stage interconnection network [58].

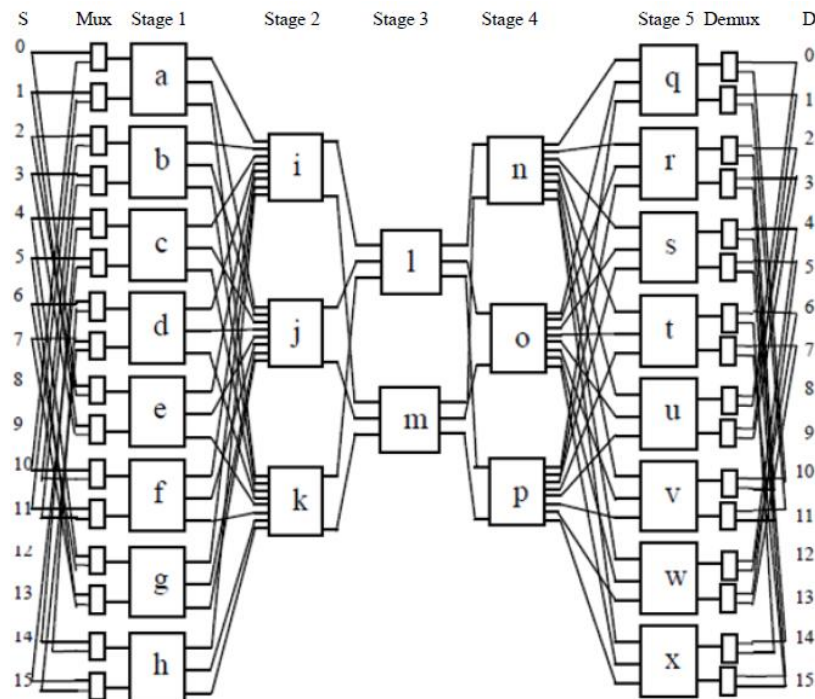


Figure 2.6. 16×16 IMABN-2.

A 16×16 IMABN-2 is shown in figure 2.6. The source address, destination address, multiplexer, and demultiplexer are represented by S, D, Mux, and Demux respectively. In IMABN-2 the size of each Mux and Demux are 2×1 and 1×2 respectively. The size of each SE in first, second, third, fourth, and the fifth stage is 2×3, 8×3, 3×3, 2×8, and 3×2 respectively. It follows irregular network topology.

In the routing process [58], if a request arrives on a SE, then it forwards it towards the given destination side. If the SE is faulty then request arrives on the non-faulty SE. There are always two possibilities, both the required SE is available and fault free otherwise it is not available for data transmission or faulty. In the first case, data packets move to the given SE to get its given destination otherwise the request will be dropped.

2.3.5 Advance Omega Network

It is a blocking MIN [16]. The size of each SE in AON is 4×4 .

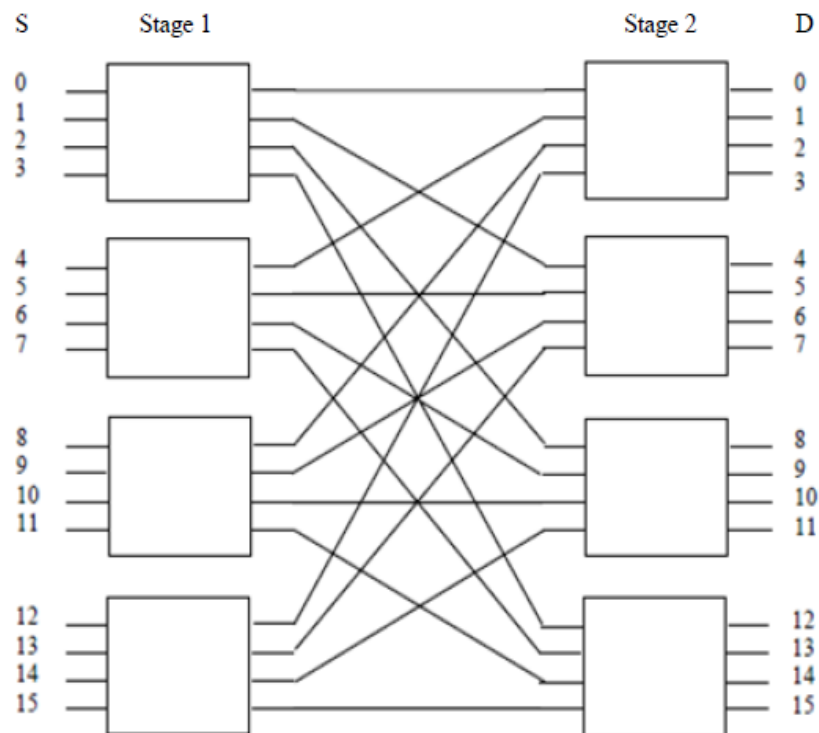


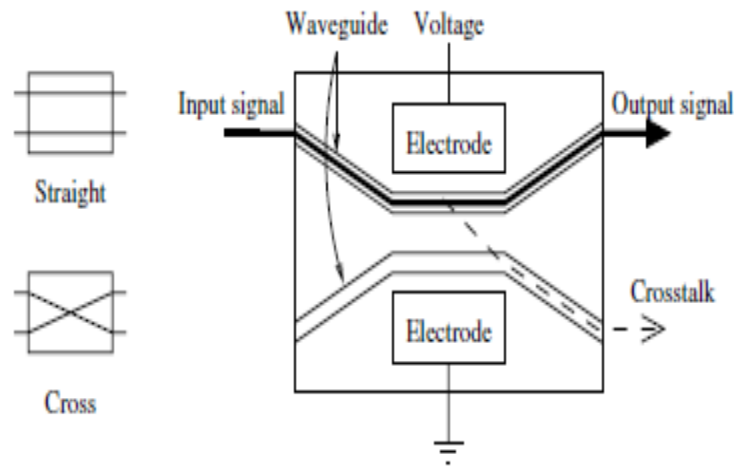
Figure 2.7. 16×16 AON.

It is the advance version of the omega interconnection network. A 2-stage interconnection network is shown in figure 2.7. In this figure, the source and destination are represented by S and D respectively. This network [16] is more fault tolerant than the omega network. There are 16 sources 16 destinations in figure 2.7. Designation tag routing is followed by AON in its routing methodology.

2.4 Crosstalk in Multi-stage Interconnection Networks

Crosstalk is the challenging problem in MINs [27, 28, 42]. The signal to noise ratio is affected because of crosstalk. It also limited the size of a network and therefore, MIN gets deprivation in terms of efficiency. It can be defined as:

“Crosstalk occurs when two signal channels interact with each other. In cross talk, a small fraction of the input signal power may be detected at another output, although the main signal is injected at the right output. For this reason, the input signal will be distorted at the output due to the loss and crosstalk introduced on the path [23]”.



. Figure 2.8. Crosstalk in an electro optic SE.

Crosstalk is shown in figure 2.8 [29]. To overcome the crosstalk problem, many approaches have been proposed in literature which is explained in the next section.

2.5 Methods to Reduce the Crosstalk in MINs

Basically, there are two approaches which are used to reduce the crosstalk in MINs i.e. space domain approach and time domain approach. These approaches are explained in the next subsections.

2.5.1 Space Domain Approach

In space domain approach (SDA), a duplicate copy of a MIN is combined with the original MIN to minimize the crosstalk problem [29, 42].

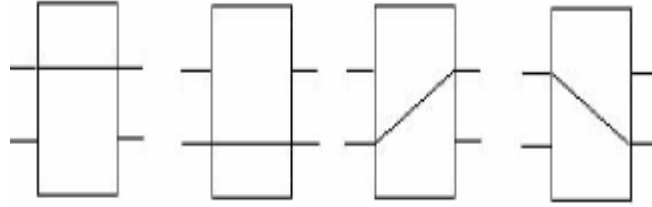


Figure 2.9. Legal passes to avoid the Crosstalk.

This method requires more than the double network hardware to obtain the crosstalk free routes. Further, extra SEs and links are used in SDA. Therefore, it can be said that the space domain is an expensive and time consuming approach. The legal passes [23] are shown in figure 2.9.

2.5.2 Time Domain Approach

The time domain approach (TDA) reduces the crosstalk problem by allowing only one source and its corresponding destination address to be active at a time within a SE in the network [28,42]. Crosstalk is treated as a conflict in TDA. This approach makes its important because of various reasons like most of the multiprocessor system use electronic processor and optical MINs.

Therefore, there is a big mismatch between the processing speeds of the network carrying optical signals [23-29]. In TDA, permutation and semi-permutation is applied to the message groups so that each group is routed in a different time slot. A combination matrix is obtained by combining the given source and destination address.

Further, message partitioning is performed based on the combination matrix. The all activity is performed so that a particular set of messages should get their given destinations in the first pass in crosstalk free environment. The other messages are passed in next passes. There are various methods for message partitioning e.g. window method [23-27], improved window method [23-29], heuristic routing algorithms [23-29]. Message separation in window method is based on the bit pattern. If the bit patterns of two or more messages are same then they will be routed in the different passes.

This method also generates a combination matrix. This matrix has a window size $M-1$, here $M=\log_2N$ and N is the size of network. In window method, the first and last columns of combination matrix are always avoided and calculations are performed on the rest of the columns. Improved window method always avoids the first window and does the calculations on rest of the windows. This method takes less time as compared to window method. In heuristic routing approach, messages can be scheduled in four ways which are as follows:

- Schedule the messages in their increasing order.
- Schedule the messages in their decreasing order.
- Schedule the messages in their lowest degree of conflict.
- Schedule the messages in their highest degree of conflict.

2.6 Conclusion

In this way, the author has explained the faulty switch problem and crosstalk problem. Both these problem creates a negative impact on the efficiency of MINs. Further, he has also given a brief description of earlier proposed methods regarding both problems. The author presents solutions against faulty switch problem in chapter 3 and 4. In chapter 5, he has given solutions against the crosstalk problem. He has mainly focused on increasing efficiency of MINs and therefore, he has proposed solutions in chapters 3, 4, and 5 towards these problems.

CHAPTER 3

COMPARATIVE ANALYSIS OF FAULT TOLERANT MINS BASED ON COST AND AVAILABILITY OF PATHS

3.1 Introduction and Motivation

Interconnection Network (IN) [59-65] plays a key role in providing communication amid processors, amid processors and memory modules. Multi-stage interconnection network (MIN) is a cost effective IN and therefore, it is used in the parallel processing system (PPS). INs [62-65] have uniform or non-uniform connection pattern and it classifies the MINS to be regular or irregular respectively. Generally, a MIN has $n=\log_2N$ stages and each stage has $N/2$ switching elements (SEs), where N is the size of the network.

Basically, 2×2 SEs are used in MINS. MINS [6, 31-35] have multipath nature and it yields a good fault tolerant capability in the network. The basic idea of having fault tolerance is to establish multiple paths between a source and destination pair and the alternate path can be used in case of faults in the primary path [22]. In order to increase the fault tolerance, many authors have proposed various network [20-22] models with their routing algorithm. However a unique solution has not yet obtained.

3.2 Related Work

In the present chapter, the author has proposed three new interconnection networks named as advance irregular augmented shuffle exchange network (AIASEN), irregular augmented shuffle exchange network-2 (IASEN-2), advance irregular alpha multi-stage interconnection network (AIAMIN) and their routing algorithms.

The architecture of AIASEN and IASEN-2 are based on previously proposed irregular augmented shuffle exchange network (IASEN). It has been observed that the AIASEN and IASEN-2 engender better fault tolerance as compared to IASEN [20]. In the same way, the architecture of AIAMIN is based on previously proposed modified alpha network (MALN) [21]. It has been observed that the AIAMIN has better fault tolerability than the MALN. The structures and routing algorithms of IASEN and MALN is already discussed in chapter 2.

3.3 AIASEN

This section gives a detail description of AIASEN.

3.3.1 Structure of AIASEN

AIASEN is a modified interconnection network with two extra stages. This network has 5 stages as shown in figure 3.1.

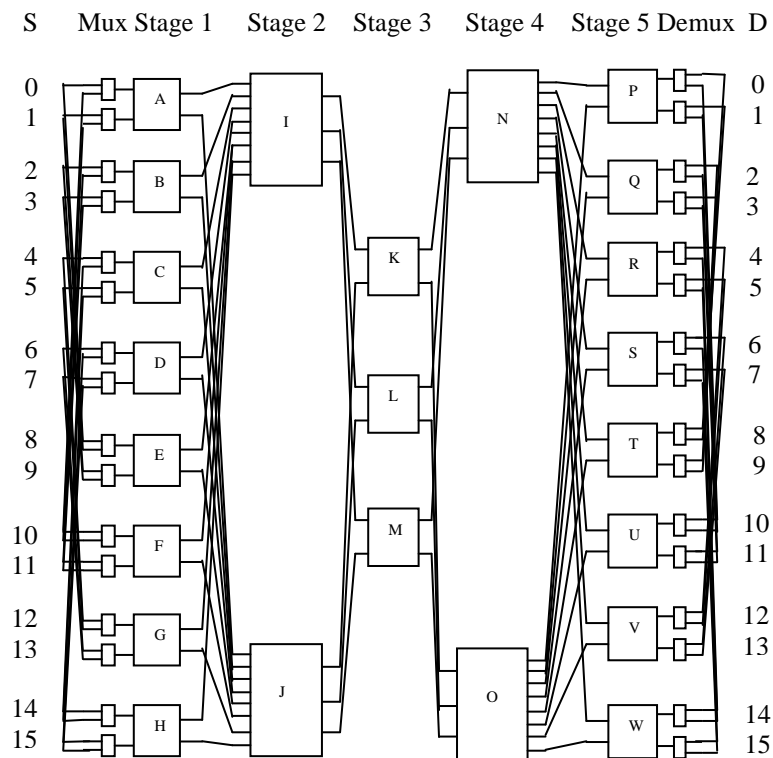


Figure 3.1. 16x16 AIASEN.

The first and last stage of the network has an equal number of SEs i.e. each stage has 8 SEs. The second and fourth stage consists of 2 SEs in each one. There are 3 SEs in the third stage. The source address, destination address, multiplexer, and demultiplexer are represented by S, D, Mux and Demux respectively. In figure 3.1, it can be observed that 16 Mux are connected at source side and 16 Demux are connected at the destination side. The size of each mux and demux is 2×1 and 1×2 respectively. In AIASEN, 2×2 is the size of each SE in first, third, and the last stage. Similarly, 8×3 and 3×8 is the size of each SE in second and fourth stage.

In figure 3.1, each source and destination address are connected with one primary SE and two secondary SEs e.g. source address 0 is connected with SEs A (primary SE), E (secondary SE) and F (secondary SE) through multiplexer and destination address 0 is connected with SEs P (primary SE), T (secondary SE) and U (secondary SE) through demultiplexer and in the same way the other source and destination addresses are connected. In this way, connectivity among the stages is clearly shown in figure 3.1.

3.3.2 Routing Algorithm of AIASEN

In the routing algorithm of AIASEN, first obtain the source and its corresponding destination address. If the SEs of first, third and fifth stage gets the request and if they are faulty then request can be sent to the other 2 SEs of that particular stage. If the SEs of second and fourth stage gets the request and if they are faulty then request can be sent to only one SE of that particular stage.

Routing Algorithm

1. Get the source and destination address sequentially.
2. SE of first stage gets the request on any of its two input links through the multiplexer. If this SE is faulty then send the request to the second SE of first stage and if the second SE is also faulty then send the request to the third SE of first stage through the multiplexer. If it is also faulty then drop

the request otherwise send the request to the appropriate SE of next stage and go to step 3.

3. SE of second stage gets the request on any of its eight input links from the SE of the first stage. If the required SE is faulty then send the request to the second SE of stage 2. If the second SE is also faulty then drop the request otherwise send the request to the appropriate SE of next stage and go to step 4.

4. SE of third stage gets the request on any of its two input links from the SE of second stage. If the required SE is faulty then send the request to the second SE of stage 3. If the second SE has also faulty then sent the request to the third SE of stage 3. If it is also faulty then drop the request otherwise send the request to the appropriate SE of next stage and go to step 5.

5. SE of fourth stage gets the request on any of its three input links from the SE of third stage. If the required SE is faulty then send the request to the second SE of stage 4. If it is also faulty then drop the request otherwise send the request to the appropriate SE of next stage and go to step 6.

6. SE of fifth stage gets the request on any of its two input links from the SE of the fourth stage. If this SE is faulty then send the request to the second SE of fifth stage and if the second SE is also faulty then send the request to the third SE of the fifth stage. If it is also faulty then drop the request otherwise go to step 7.

7. Send the request to its destination address through the demultiplexer.

8. End.

3.3.2.1 Proof of Correctness

To prove the algorithm the author has taken an example which is as follows:

Example 1: It is assumed that the source and destination addresses are 0 and 3 respectively. In this example, the author has considered three cases which are as follows:

Case1: When SEs are not faulty in every stage.

Algorithmic Step1: Get the source and its destination address; here the source and destination address is 0 and 3 respectively.

Algorithmic Step2: In first stage, all the SEs are not faulty therefore, SE A will get the request through a multiplexer and send the request to the SE of the next stage.

Algorithmic Step3: In the second stage, SE I will receive the request and it sends the request to the SE of the next stage.

Algorithmic Step4: In the third stage, SE K will receive the request and it sends the request to the SE of the next stage.

Algorithmic Step5: In the fourth stage, SE N will receive the request and it sends the request to the SE of the next stage.

Algorithmic Step6: In the fifth stage, SE Q will receive the request and it sends the request to its destination address i.e. 3 through demultiplexer.

Algorithmic Step7: End.

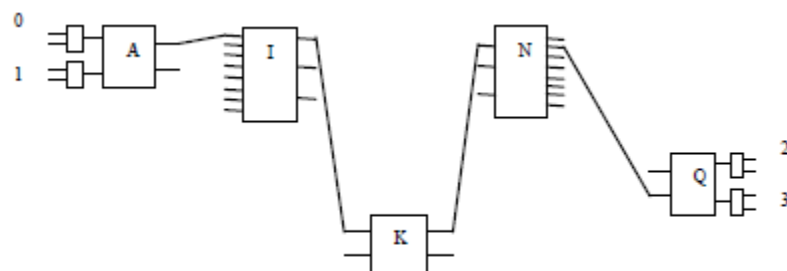


Figure 3.2. Sending request from 0 to 3.

In figure 3.2, the source address 0 is sending request to destination address 3 through the intermediate SEs as explained in all the algorithmic steps. The request sending process is shown in figure 3.2.

Case2: When one SE of each stage is faulty.

Algorithmic Step1: Get the source and its destination address; here the source address is 0 and its destination address is 3.

Algorithmic Step2: It is supposed that the SE A in the first stage is faulty then sending the request to SE E through the multiplexer. SE E will send the request to the SE of the next stage.

Algorithmic Step3: It is supposed that the SE I in the second stage is faulty then SE E will send the request to SE J. Further, SE J will send the request to the SE of the next stage.

Algorithmic Step4: It is supposed that the SE K in the third stage is faulty then then SE J will send the request to SE L. Further, SE L will send the request to the SE of the next stage.

Algorithmic Step5: It is supposed that the SE N in fourth stage is faulty then then SE L will send the request to SE O. Further, SE O will send the request to the SE of the next stage.

Algorithmic Step6: It is supposed that the SE Q in fifth stage is faulty then SE O will send the request to SE T. Further, SE T will send the request to its destination address through demultiplexer.

Algorithmic Step7: End.

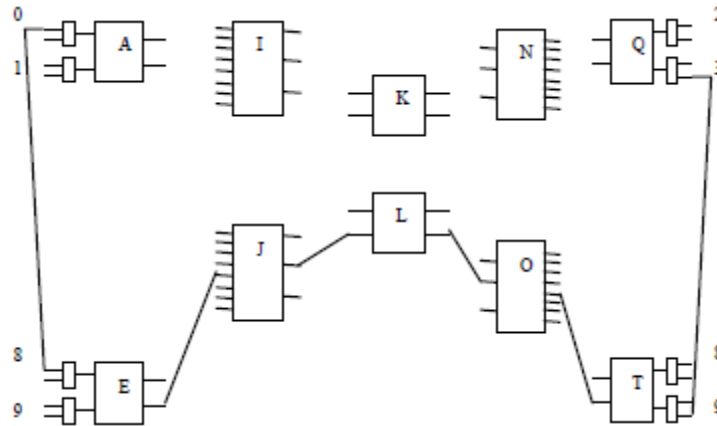


Figure 3.3. Sending request from 0 to 3 through intermediate SEs.

In figure 3.3, the source address 0 is sending request to destination address 3 through the intermediate SEs as explained in all the algorithmic steps. SEs A, I, K, N, and Q are faulty.

Case3: When two SE of stage 1, 3 and 5 are faulty and one SE of stage 2 and 4 are faulty.

Algorithmic Step1: Get the source and its destination address; here again the source address is 0 and its destination address is 3.

Algorithmic Step2: It is supposed that the SE A and E in the first stage are faulty then sending the request to SE F through the multiplexer. Further, SE F will send the request to the SE of the next stage.

Algorithmic Step3: It is supposed that the SE J in second stage is faulty then SE F will send the request to SE I. Further, SE I will send the request to the SE of the next stage.

Algorithmic Step4: It is supposed that the SE K and L in the third stage are faulty then SE I will send the request to SE M. Further, SE M will send the request to the SE of the next stage.

Algorithmic Step5: It is supposed that the SE O in fourth stage is faulty then SE M will send the request to SE N. Further, SE N will send the request to the SE of the next stage.

Algorithmic Step6: It is supposed that the SE Q and T in fifth stage are faulty then SE N will send the request to SE U. Further, SE U will send the request to its destination address through demultiplexer.

Algorithmic Step7: End.

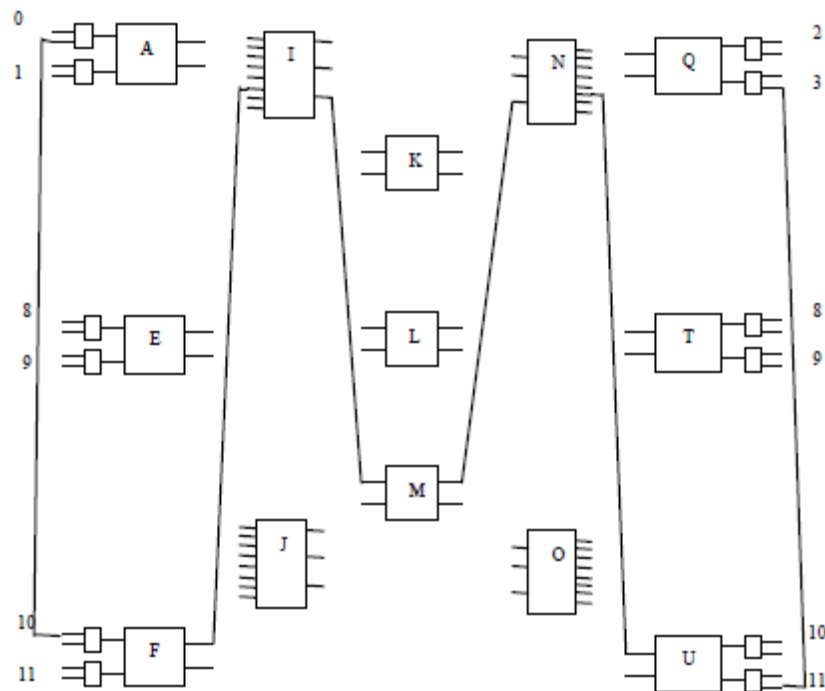


Figure 3.4. Sending request from 0 to 3 through intermediate SEs.

In figure 3.4, the source address 0 is sending request to destination address 3 through the intermediate SEs as explained in all the algorithmic steps. The SEs A, E, J, K, L, O, Q and T are faulty. In this way, data packets can be sent from any source to any destination through AIASEN in faulty and non-faulty conditions.

3.3.3 Path Availability

The IASEN provides only eight paths between a source-destination pair and the proposed IN has more number of paths between a source-destination pair [34, 36]. It is explained in the following theorem:

Theorem1: There are at least twelve paths between every source and destination address.

Proof1:

Figure 3.5 shows the possible paths between source 2 and destination 1. The generation of paths depend on the middle stages of the network. The SEs of first stage are allied with 2 SEs of second stage. The SEs of second stage are linked with 3 SEs of third stage. Further, the SEs of third stage are connected with 2 SEs of fourth stage. Similarly, the SEs of fourth stage are connected with the SEs of fifth stage.

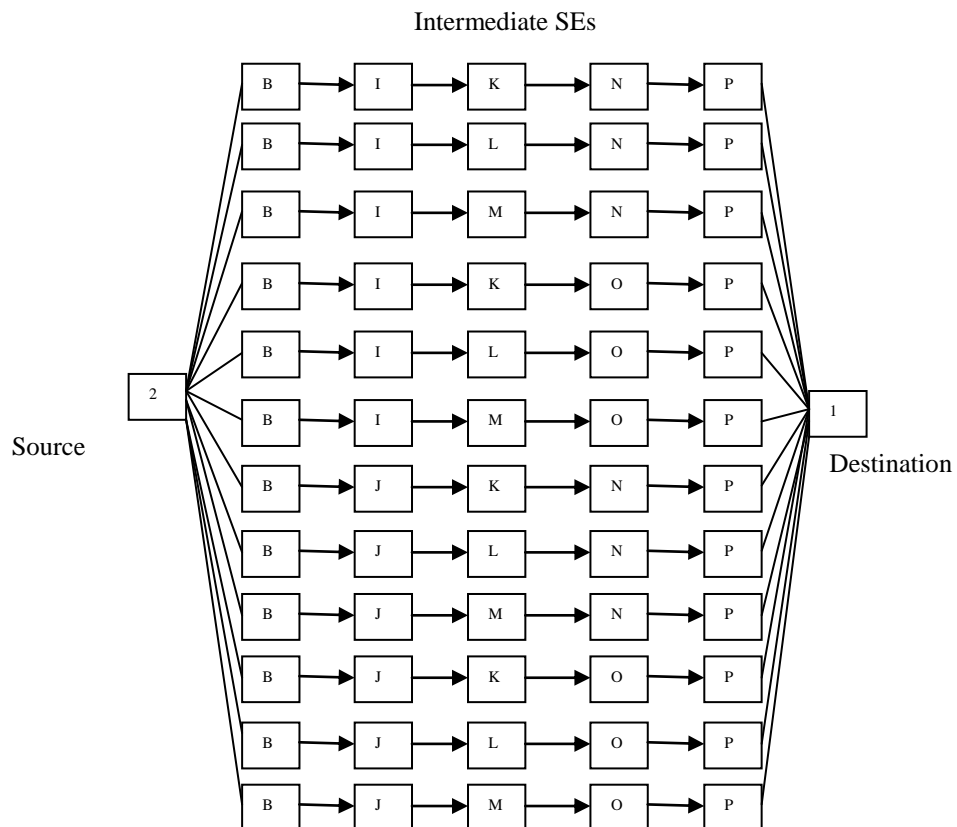


Figure 3.5. Sending request from 2 to 1 through intermediate SEs.

Therefore, it is concluded that all the paths are generated through second, third and fourth stages and the number of paths will be:

$$\Rightarrow (\text{Number of SEs in Stage2}) \times (\text{Number of SEs in Stage3}) \times (\text{Number of SEs in Stage4})$$

$$\Rightarrow (2) \times (3) \times (2)$$

$$\Rightarrow 12$$

So, this fact prove the theorem.

3.3.4 Cost Analysis

The cost of the network is analyzed on the basis of the components which are used in the proposed interconnection network [20, 21, 34]. The components of the proposed interconnection network are SEs of different size, multiplexers and demultiplexers. The cost of any hardware component is proportional to the number of gate counts, which are used in it and therefore the cost of proposed AIASEN is calculated as follows:

Total number of 2×2 SEs=19, cost=76,

Total number of 8×3 SEs=2, cost=48,

Total number of 3×8 SEs = 2, cost = 48,

Total number of multiplexers =16, cost = 32,

Total number of demultiplexers =16, cost = 32.

Hence, the total cost of AIASEN is 236 units. The total cost of IASEN [20] is 268 units and it is an expensive interconnection network with a limited number of paths between every pair of source and destination address and with limited fault tolerance capability. The proposed interconnection network is a fault tolerant network with more alternate paths and with full accessibility.

3.4 IASEN-2

This section gives a detail description of IASEN-2.

3.4.1 Structure of IASEN-2

Irregular augmented shuffle exchange network-2 is derived from Irregular Augmented Shuffle Exchange Network. The source address, destination address, multiplexer, and demultiplexer are represented by S, D, Mux and Demux respectively. In figure 3.6, it can be observed that 16 Mux are connected at source side and 16 Demux are connected at the destination side. There are 16 sources and 16 destinations in IASEN-2. It has four stages. There are 8 SEs in first and last stage in each one.

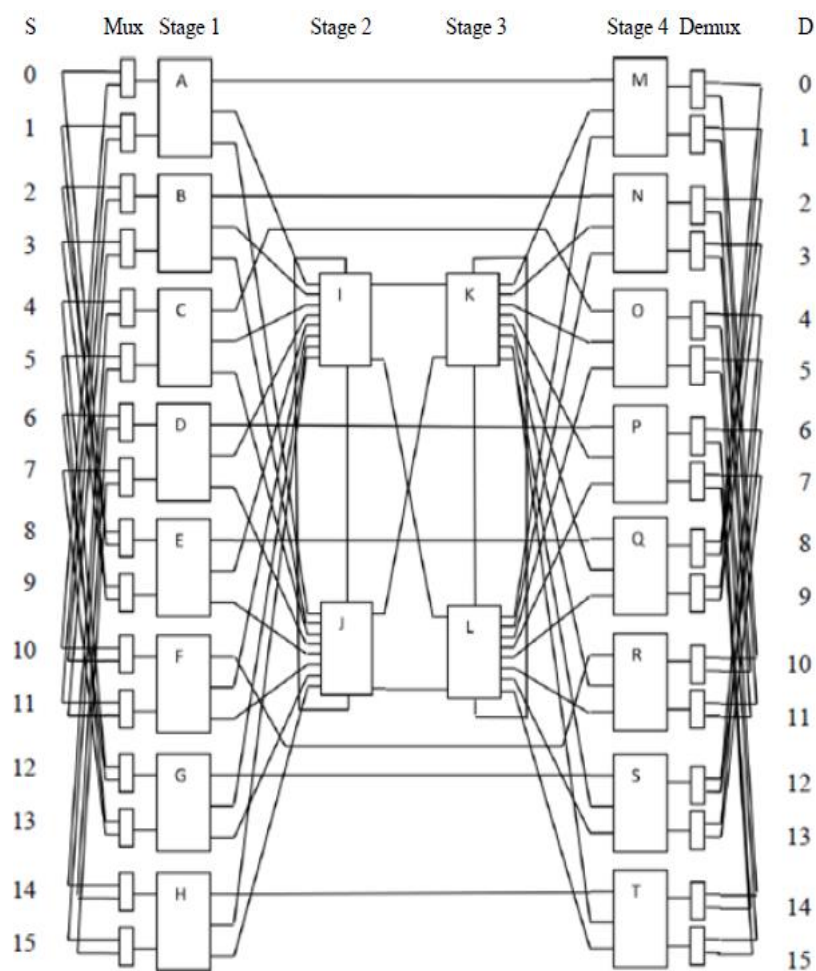


Figure 3.6. 16×16 IASEN-2.

In first and last stage size of each SE is 2×3 and 3×2 respectively. Second and third stage consists of 2 SEs in each one. In IASEN-2, 9×3 and 3×9 are the size of SEs of second and third stage respectively. In IASEN-2, the size of Mux is 2×1 and Demux is 1×2 . Two Mux are connected with each SE of the first stage and two Demux are connected with each SE of the fourth stage

as shown in figure 3.6. The SEs of the first stage are connected with SEs of second stage, SEs of the second stage are connected with SEs of the third stage, SEs of third stage are connected with SEs of the fourth stage as shown in figure 3.6.

The SEs A and B are connected with SEs E and F through a Mux. In the same way, the SEs C and D are also connected with SEs G and H. The SEs M and N are connected with SEs Q and R through Demux. In the same way, the SEs O and P are also connected with SEs S and T. In second and third stage the SEs are connected with each other through auxiliary links. These links are shown in figure 3.6.

3.4.2 Routing Algorithm of IASEN-2

In IASEN-2, two algorithms are proposed. SA and DA are the source and destination addresses. If source and destination addresses are different then it will follow algorithm1. According to algorithm1, if there are more than two faulty SEs in first or last stage, then drop the request otherwise send the request to the appropriate SE of that stage through a Mux. In second and third stage, if there is more than one SE is faulty then drop the request otherwise send the request to the appropriate SE of that stage. Here SE_I , SE_J , SE_K and SE_L are SEs I, J, K and L respectively.

Algorithm1:Routing_IASEN-2

1. Begin.
2. Get SA and its corresponding DA and send the request to the appropriate SE of first stage through Mux and go to step 3.
3. The SE of first stage gets the request on any of its two input links. If this SE is faulty then request will be received by first alternate SE of stage1 and if it is faulty then request will be received by second alternate SE of first

stage through a Mux. If it is also faulty then drop the request otherwise send the request to the SE_I and go to step 4.

4. SE_I gets the request on any of its nine inputs links. If it is faulty then request will be received by SE_J . If it is faulty then drop the request otherwise send the request to SE_K and go to step 5.

5. SE_K gets the request on any of its three inputs links. If it is faulty then request will be received by SE_L . If it is also faulty then drop the request otherwise send the request to the appropriate SE of fourth stage and go to step 6.

6. SE of fourth stage gets the request on any of its three input links from SE_K or SE_L . If it is faulty then request will be received by first alternate SE of fourth stage and if it is faulty then request will be received by second alternate SE of the fourth stage through Demux. If it is also faulty then drop the request otherwise go to step 7.

7. Send the request to its destination address through Demux.

8. End.

In algorithm2, the ASE_1 and ASE_4 are appropriate SE of first stage and appropriate SE of the fourth stage respectively. According to algorithm2, if source and destination addresses are same and both ASE_1 and ASE_4 are not faulty then directly send the request to its destination through Demux otherwise follow algorithm1.

Algorithm2:Routing_IASEN-2

```
If (SA and DA are same)
{
If ( $ASE_1$  and  $ASE_4$  are not faulty)
{
Send request from  $ASE_1$  to  $ASE_4$ ;
}
```



```

else
{
Algorithm1;
}
}
else
{
Algorithm1;
}

```

3.4.2.1 Proof of Correctness

In the proposed network, it is shown that the SEs of all stages have the auxiliary links. Each Source has the connectivity with 3 SEs of first stage. Similarly, each destination has the connectivity with 3 SEs of final stage. Therefore, the first and final stage can sustain 2 faulty SEs simultaneously. Further, stage 2 and 3 has 2 SEs in each one therefore, each stage can tolerate the single faulty SE simultaneously. This fact is shown in example 3.

Example 3: It is supposed that the source and destination addresses are 6 and 5 respectively. In this example, the source and destination address are different and therefore the author has followed the algorithm1. Here, two cases are taken into consideration which are as follows:

Case1: When SEs are non-faulty in every stage.

1. Begin.
2. According to the algorithm, the source and destination address is 6 and 5 respectively. Now, the request is sent to SE_D through Mux and algorithmic step 3 will be followed.
3. In this step, SE_D will get the request on its first input link. Now, the request is being sent to the SE_I and algorithmic step 4 will be followed.

4. SE_I will get the request on its fourth input link from the SE_D and it sends the request to SE_K and it follows the step 5 of the algorithm.
5. Now SE_K gets the request on its first input link from SE_I and it send the request to SE_O and it follows the step 6 of the algorithm.
6. Further, SE_O of fourth stage gets the request on its second input link from the SE_K and it follows the step 7 of the algorithm.
7. The request is sent to its destination 5 through Demux.
8. End.

Case2: When 1 SE is faulty in every stage. Suppose SEs D, I, K and O are faulty in every stage. Now, according to the algorithm:

1. Begin.
2. As mentioned in the algorithm, the source address is 6 and destination address is 5. Here, algorithmic step 3 will be followed.
3. It is known that SE_D is faulty and therefore, the request will be sent to the first alternate SE of stage1 i.e. to SE_G through a Mux. SE_G will get the request on its first input link. Now, the request will be sent to the SE_I of second stage and algorithmic step 4 will be followed.
4. It is known that SE_I is faulty and therefore, SE_J will get the request on its seventh input link from the SE_G and it sends the request to SE_K of third stage and follows the step 5 of the algorithm.
5. It is mentioned that SE_K is faulty and therefore, SE_L will get the request on its second input link from SE_J and it sends the request to appropriate SE_O and follows the step 6 of the algorithm.

6. It is mentioned that SE_O is faulty and therefore, SE_S will get the request on its third input link from SE_L and now it will follow the step 7 of the algorithm.

7. In the next step, request is sent to its destination 5 through Demux.

8. End.

In this way, data packets can be sent from any source to any destination through IASEN-2 in faulty and non-faulty conditions.

Theorem2:

There are atleast sixteen paths exist between every source and destination.

Proof2:

The middle stages of a network generates the alternate paths. Here $SE-I$ has the 4 paths i.e. I-K, I-L, I-J-K, and I-J-L. Similarly, $SE-J$ also has the 4 paths i.e. J-K, J-L, J-I-K, J-I-L. Here, I, J, K, L are the SE. Therefore, the total number of alternate path will be $4 \times 4 = 16$ paths. To prove the theorem, the author has taken an example which is as follows:

Example 4: It is supposed that source address is 6 and destination address is 5. Now the possible paths between these addresses are as follows:

Path1: D-I-K-O

Path2: D-I-L-O

Path3: D-I-K-L-O

Path4: D-I-L-K-O

Path5: D-I-J-K-O

Path6: D-I-J-L-O

Path7: D-I-J-L-K-O

Path8: D-I-J-K-L-O

Path9: D-J-K-O

Path10: D-J-L-O

Path11: D-J-K-L-O

Path12: D-J-L-K-O
Path13: D-J-I-K-O
Path14: D-J-I-L-O
Path15: D-J-I-L-K-O
Path16: D-J-I-K-L-O

Here D, I, J, K, L and O are the SEs. It proves that IASEN-2 has sixteen paths between every pair of source and destination.

3.4.3 Cost Analysis

The cost [20, 21] of the proposed interconnection network can be calculated as follows:

Total number of 2×3 SEs = 8, cost = 48,
Total number of 9×3 SEs = 2, cost = 54,
Total number of 3×9 SEs = 2, cost = 54,
Total number of 3×2 SEs = 8, cost = 48,
Total number of 2:1 multiplexers = 16, cost = 32,
Total number of 1:2 demultiplexers = 16, cost = 32.

Therefore, 268 units is the cost of IASEN-2. The cost of IASEN [20] is also 268 units. There are only eight paths in IASEN whereas IASEN-2 has 16 paths between every source and destination with better fault tolerance capacity.

3.5 AIAMIN

This section gives a detail description of AIAMIN.

3.5.1 Structure of AIAMIN

The AIAMIN is an irregular MIN and it has 16 sources and 16 destinations. This network has 4 stages. In the figure 3.7, the S represents the source address, Mux represents the Multiplexer, and Demux represents the demultiplexer and D represent the destination address. In first stage, it has 8 switch-

es of size 2×3 . The first switch is named as A, second switch named as B, and in the same way the author has given names to all the switches of the first stage. Each switch of the first stage is connected with a 2×1 Mux for each input link. In first stage each source address is connected with three switches through Mux e.g. the source address 0 is connected with SEs A, E and F.

In the second stage, this network has 3 switches of size 8×2 and the first switch of this stage is named as I, second switch named as J, and third switch named as K. The switches of the first stage are connected to the switches of second stage through their output links, e.g. SE A is connected with SEs I, J and K through its output link. In third stage, this network has two switches of size 3×8 and the first switch of this stage is named as L and second switch named as M.

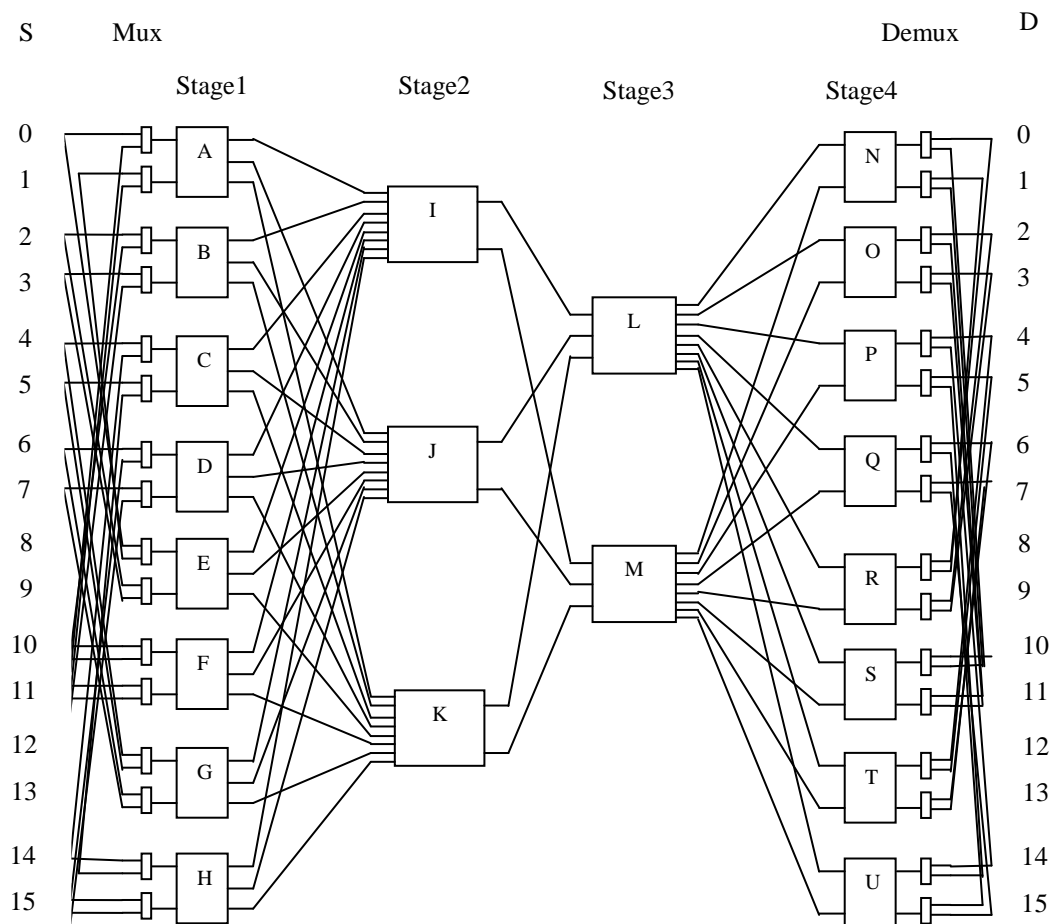


Figure 3.7. 16×16 AIAMIN.

The switches of the second stage are connected to the switches of third stage through their output links, e.g. SE I is connected with SEs L and M through its output link. In fourth stage, this network has 8 switches of size 2×2 and the first switch of this stage is named as N, second switch named as O, and equally the author has given names to all the switches. Each switch of fourth stage is connected with a 1×2 Demux for each output link.

In the fourth stage, each destination address is connected with three switches through Demux e.g. the destination address 0 is connected with N, R and S. The switches of third stage are connected to the switches of the fourth stage through their output links, e.g. SE L is connected to SEs N, O, P, Q, R, S, T and U through its output link. In this way, connectivity among the stages is clearly shown in figure 3.7.

3.5.2 Routing Algorithm of AIAMIN

The routing algorithm is designed in such a way that it can send data between any source and destination address in both conditions, i.e. in the absence of faults and in the presence of multiple faults. In the first step, the source and its destination address are selected. Now send the request to the suitable SE of the first stage, if it is faulty then followed step 7 otherwise go to step 4. In the next step, request is being sent from the SE of the first stage to the SE of second stage. If the SEs of the second stage are faulty then algorithmic step 8 will be followed, otherwise algorithmic step 5 will be followed.

Further, a request will be sent from the SE of the second stage to the SE of the third stage, if the SEs of third stage are faulty then algorithmic step 9 will be followed, otherwise algorithmic step 6 will be followed. In the next step, request will be sent from the SE of third stage to the SE of the fourth stage. In this way, data packets can be sent from any source to any destination.

1. Start.
2. Get the source and its destination address.
3. Send the request to the appropriate SE of stage1. If SE of stage1 is faulty then go to step7, otherwise go to step4.
4. Now send the request from the SE of the first stage to the first SE i.e. SE I of stage 2. If the SE I of stage 2 is faulty then go to step8, otherwise go to step5.
5. Send the request from the SE of the second stage to the SE L of stage 3.If the SE L is faulty then go to step 9, otherwise go to step 6.
6. Receive the request on SE L or M and go to step 10.
7. Now send the request to the first auxiliary SE of stage 1 and go to step 4, if it is faulty then sending the request to the second auxiliary SE of stage1 and go to step 4, if this SE is also faulty then drop the request and go to step11.
8. Now send the request to SE J and go to step 5, if SE J is faulty then send the request to SE K and go to step 5, if SE K is also faulty then drop the request and go to step11.
9. Now send the request to SE M and go to step 6, if SE M is faulty then drop the request and go to step11, otherwise go to step 6.
- 10.Send the request from SE L or M to the appropriate SE i.e. SE that contain the destination address.
11. End.

3.5.2.1 Proof of Correctness

To prove the algorithm the author has taken an example which is as follows:

Example 5: It is assumed that the source and destination address is 0 and 2 respectively. Here, the author has considered three cases which are as follows:

Case 1: When SEs of stage 1, stage 2, and stage 3 are not faulty.

Algorithmic Step1: Start.

Algorithmic Step 2: Get the source and its destination address, here the source address is 0 and its destination address is 2.

Algorithmic Step 3: Request is sent to the SE A in stage 1.

Algorithmic Step 4: Request is sent to SE I from SE A. It is clearly shown in figure 3.8

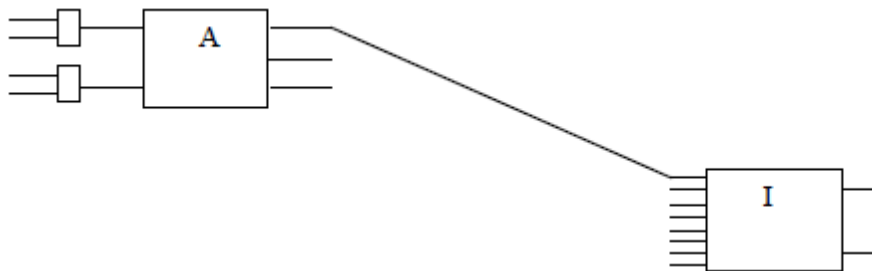


Figure 3.8. Sending request from SE A to SE I.

Algorithmic Step 5: Request is sent to SE L from SE I. It is shown in figure 3.9.

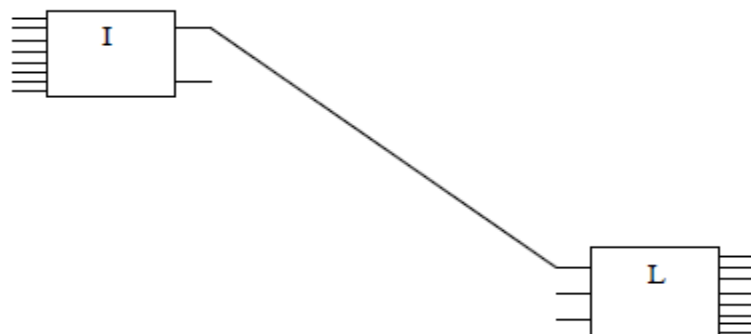


Figure 3.9. Sending request from SE I to SE L.

Algorithmic Step 6: Switch L will receive the request and algorithmic step 10 will be followed.

Algorithmic Step 7: Request is sent to SE O from SE L. It is shown in figure 3.10.

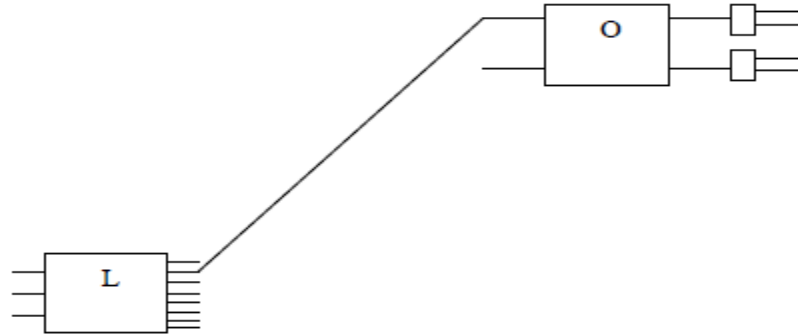


Figure 3.10. Sending request from SE L to SE O.

Algorithmic Step8: End.

In this way, the data packets can be sent from source address 0 to destination address 2 through Demux.

Case 2: When 1 SE of stage 1, 1 SE of stage 2 and 1 SE of stage 3 are faulty. It is supposed that the SEs A, I, and L are faulty.

Algorithmic Step1: Start.

Algorithmic Step2: Get the source and its destination address, here the source address is 0 and its destination address is 2.

Algorithmic Step3: It is assumed that SE A is faulty, therefore, algorithmic step 7 will be followed and a request will be sent to first auxiliary SE i.e. SE E through auxiliary links. The auxiliary links are shown by dotted lines in figure 3.11.

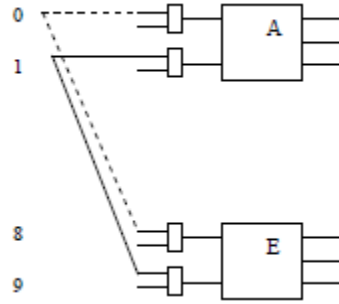


Figure 3.11. Sending request from source 0 to SE E.

Algorithmic Step4: It is mentioned that SE I is faulty therefore step 8 will be followed. In this step, the request will be sent from SE E to SE J as shown in figure 3.12.

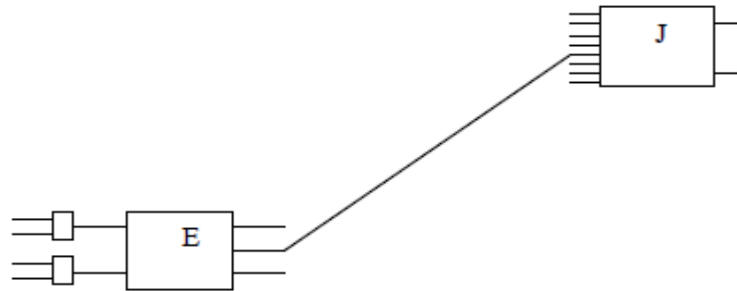


Figure 3.12. Sending request from SE E to SE J.

Algorithmic Step5: It is assumed that SE L is faulty, therefore step 9 will be followed. In this step the request will be sent from SE J to SE M as shown in figure 3.13.

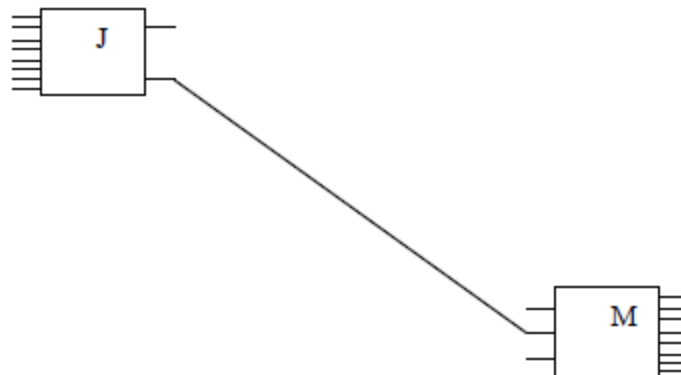


Figure 3.13. Sending request from SE J to SE M.

Algorithmic Step6: SE M will receive the request and algorithmic step 10 will be followed.

Algorithmic Step7: Request is sent to SE O from SE M as shown in figure 3.14.

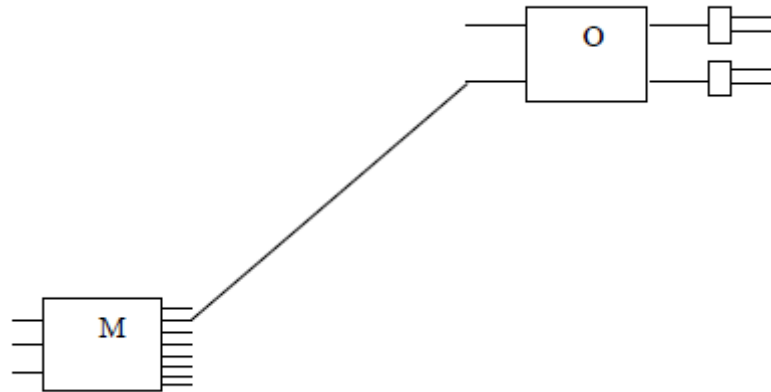


Figure 3.14. Sending request from SE M to SE O.

Algorithmic Step8: End.

In this way data packets can be sent source address 0 to 2 in the case of one faulty SE in first, second and third stage.

Case 3: When 2 SEs of stage 1, the 2 SEs of stage2 and 1SE of stage 3 are faulty. It is supposed that the SEs A, E, I, J and L are faulty

Algorithmic Step1: Start.

Algorithmic Step2: Get the source and its destination address, here the source address is 0 and its destination address is 2.

Algorithmic Step3: It is assumed that SE A and SE, E are faulty, therefore, algorithmic step 7 will be followed and a request will be sent to second auxiliary SE i.e. SE F. The auxiliary links are shown by dotted lines in figure 3.15.

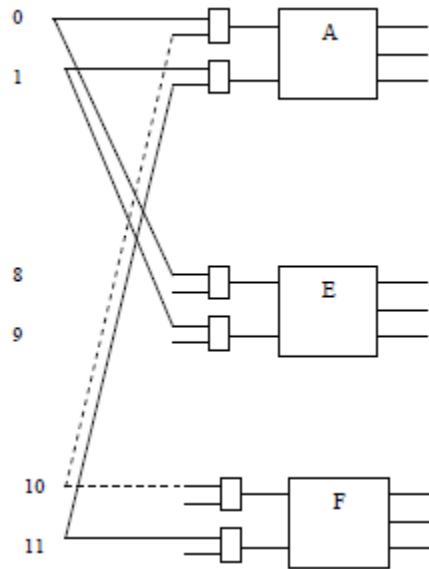


Figure 3.15. Sending request from source 0 to F.

Algorithmic Step4: It is mentioned that SE I and SE J are faulty, therefore algorithmic step 8 will be followed. In this step, the request will be sent from SE F to SE K as shown in figure 3.16.

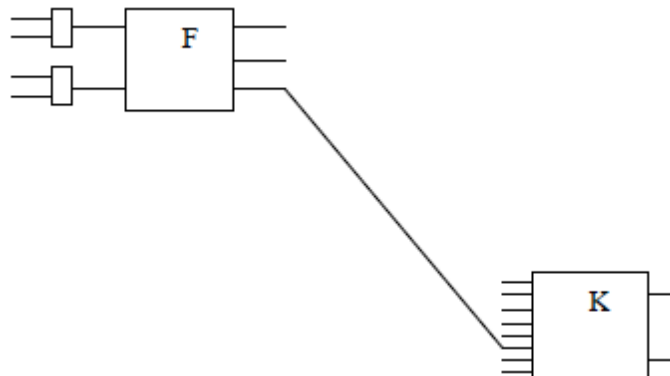


Figure 3.16. Sending request from SE F to SE K.

Algorithmic Step5: It is mentioned that SE L is faulty, therefore algorithmic step 9 will be followed. In this step, the request will be sent from SE K to SE M as shown in figure 3.17.

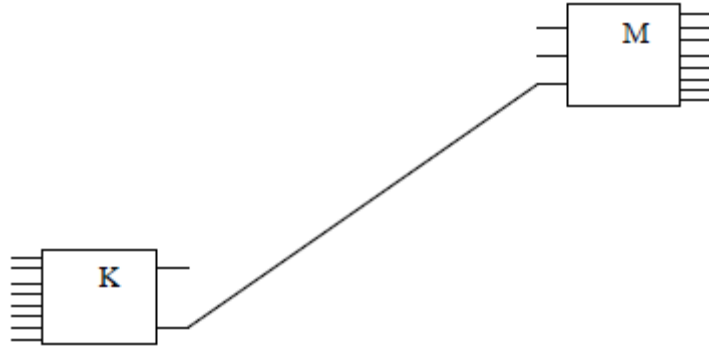


Figure 3.17. Sending request from K to M.

Algorithmic Step6: SE M will receive the request and algorithmic step 10 will be followed.

Algorithmic Step7: Request is sent to SE O from SE M as shown in figure 3.14.

Algorithmic Step8: End.

In this way, data packets can be sent from any source to any destination through AIAMIN in faulty and non-faulty conditions. This is the complete explanation of the proposed routing algorithm of the proposed network model.

3.5.2.2 Path Availability

Path availability explains [34, 36] that how many paths are available between the two nodes, i.e. between a source node and destination node. The theorem1 and 2 shows that the proposed network model has more number of paths as compared to the existing modified ALN.

Theorem3: There are atleast six alternate paths between every pair of source and destination in AIAMIN.

Proof3:

To prove the theorem, the author has taken an example which is as follows:

Example 6: It is supposed that the source address is 2 and destination address is 12 then the possible number of alternate paths between SE B and SE T are as follows:

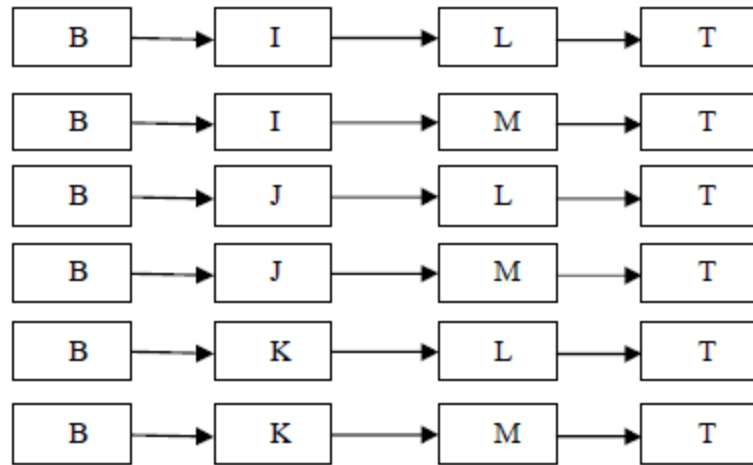


Figure 3.18. Availability of paths between SEs B and T.

From figure 3.18, it is clear that there are six alternate paths between B and T and therefore, the above shown paths prove the theorem.

Theorem4: If the SEs of stage 3 are connected through the auxiliary links then there are at least twelve alternate paths between every pair of source and destination in AIAMIN.

Proof4:

In order to get, the more alternate path the SEs of third stage should be connected through the auxiliary links. These links are shown by dotted lines in figure 3.19.

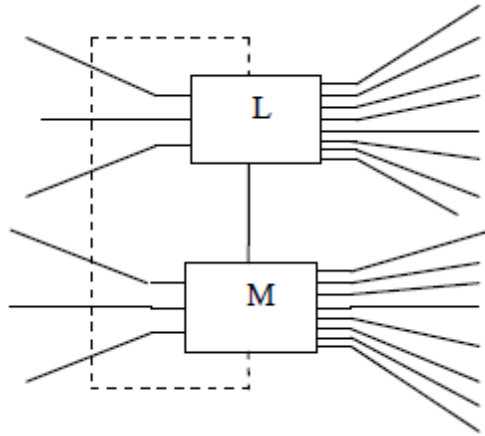


Figure 3.19. AIAMIN with auxiliary links.

Now, It is supposed that the source address is 2 and destination address is 12 then the six alternate paths will be same as explained in theorem1 and another six alternate paths are shown in figure 3.20.

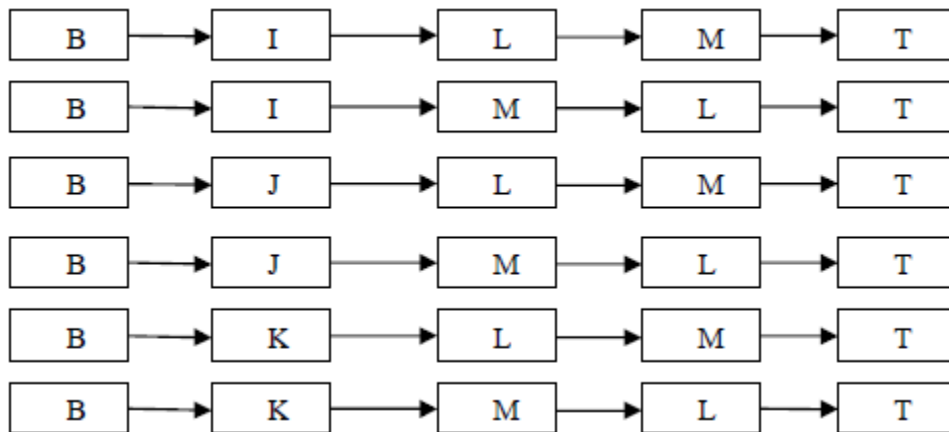


Figure 3.20. Availability of extra alternate paths between SEs B and T.

The above shown paths prove the theorem.

3.5.3 Cost Analysis

The cost [21, 34] of AIAMIN is calculated in both conditions, i.e. when it does not have the auxiliary links:

The total number of 2×3 SEs=8, cost=48,

The total number of 8×2 SEs=3, cost=48,

The total number of 3×8 SEs=2, cost=48,

The total number of 2×2 SEs=8, cost=32,
Total number of $2:1$ multiplexers=16, cost=32,
Total number of $1:2$ demultiplexers=16, cost=32.

The total cost of the AIAMIN is 240 units.

If it has auxiliary links then this network will have 2 SEs of 4×9 sizes, instead of 3×8 sizes as shown in figure 3.19 and therefore, the cost of this network will be 264 units. The total cost of MALN [21] is 240 units and it is a single switch fault tolerant network and the proposed network is 2-switch fault tolerant in some stages with more alternate paths in both cases i.e. with auxiliary links or without auxiliary links. If auxiliary links of the network are used then it will be little costly as compared to the MALN and it will provide more alternate paths as explained in theorem 2. If auxiliary links of the network are not used, then the cost of the proposed network and MALN will be same.

3.6 Conclusion and Future Scope of Work

In this chapter, the author has proposed three new fault tolerant network models named as advance irregular augmented shuffle exchange network, irregular augmented shuffle exchange network-2, and advance irregular alpha multi-stage interconnection network. The explanation of routing algorithms and theorems of AIASEN, IASEN-2, and AIAMIN show that all the proposed networks are good in terms of fault sustainability and cost.

This chapter proves that all the proposed networks have more number of alternate paths with full accessibility at minimum cost as compared to the previously proposed MINs. The main advantage of the proposed MINs that they can establish a good communication between any source and destination address even in the presence of multiple faults in the network. As for the future consideration, the author wants to design new connection pattern on the proposed MINs so that both the networks can perform efficiently in crucial faulty situations.

CHAPTER 4

COMPARATIVE ANALYSIS OF FAULT TOLERANT MINS BASED ON THE VARIOUS PERFORMANCE FACTORS

4.1 Introduction

The applications of interconnection networks (INs) are the on chip networks, storage area networks, local area networks, wide area networks and many other networks [1-6]. IN provides the connectivity amid various components of a network, e.g. connectivity between processor and memory. Multi-stage interconnection networks (MINS) are the prominent category of INs.

MINS show their eminence since it provides high speed and reliability to all parallel and telecommunication applications. Omega network [16-19], advance omega network [16], and clos network [52] are the well-known example of MINS. In general, there are N inputs and N outputs in a MIN which are connected via a number of switching stages. There are various issues which affect the efficiency of a MIN. In this chapter, the author has considered the issue of fault tolerance since it is the most extensively investigated topic in MINS. It can be understood as follows:

“The basic idea for fault-tolerance is to provide multiple paths between source and destination so that alternate paths can be used in case of faults [22].”

Further, the faulty situations in MINS create problems for the data packets which are to be transmitted from the given source to the given destination. Here the term “faulty situations” refers to the faulty switching elements (SEs) in the network. Further, the data transmission process can be based on following three methods:

- Message unicasting
- Message multicasting
- Message broadcasting

If the data packets are transmitted from a single source to a single destination, then it comes in the message unicasting method. If the data packets are transmitted from a source to an arbitrary number of destinations, then it comes in message multicasting method. If the data packets are transmitted from a source to all given destinations, then it comes in message broadcasting method. In this chapter, the author has applied message broadcasting technique to analyze the performance of proposed MINs.

4.2 Related Work

In this chapter, the author has proposed two new interconnection networks named as irregular advance omega network (IAON) and irregular augmented shuffle exchange network-3 (IASEN-3) and their routing algorithms. The architecture of IAON and IASEN-3 are based on previously proposed advance omega network (AON) [16] and irregular augmented shuffle exchange network-2 (IASEN-2). The author compares the performances of MALN [21], IASEN-2, IAON and IASEN-3 in non-faulty and single switch faulty conditions.

Single switch faulty condition means network has one faulty SE in each stage simultaneously. It has been observed that the IAON is better than the MALN and IASEN-2 in single switch faulty and non-faulty conditions. Further, when it compared with IASEN-3 then it is observed that IASEN-3 produce better results than the IAON. However, IAON shows double switch fault tolerability whereas IASEN-3 shows single switch fault tolerability. Double switch fault tolerant means the network can sustain 2 faulty SEs in each stage simultaneously. Further, the structures and routing algorithms of MALN and IASEN-2 is already discussed in chapter 2 and 3 respectively. In table 4.1 and 4.2, various symbols and their meaning are defined which are used throughout the chapter.

Symbol	Meaning of Symbol
First_Alternate_SE ₁	First alternate SE of stage1
Second_Alternate_SE ₁	Second alternate SE of stage1
First_Alternate_SE ₃	First alternate SE of stage3
Second_Alternate_SE ₃	Second alternate SE of stage3
SE _{p1}	Primary SE of stage 1
SE _{p3}	Primary SE of stage 3
F	Faulty
N	Size of Network
BW _{MALN}	Bandwidth of MALN
BW _{IASEN-2}	Bandwidth of IASEN-2
BW _{IAON}	Bandwidth of IAON
BW _{IASEN-3}	Bandwidth of IASEN-3
PA _{MALN}	Probability of acceptance of MALN
PA _{IASEN-2}	Probability of acceptance of IASEN-2
PA _{IAON}	Probability of acceptance of IAON
PA _{IASEN-3}	Probability of acceptance of IASEN-3
TT	Transmission Time
TT _{i_N_MALN}	TT of MALN of network size N in non-faulty condition
TT _{i_N_IASEN-2}	TT of IASEN-2 of network size N in non-faulty condition
TT _{i_N_IAON}	TT of IAON of network size N in non-faulty condition
TT _{i_N_IASEN-3}	TT of IASEN-3 of network size N in non-faulty condition
TT _{faulty_MALN}	TT of MALN of network size N in faulty condition
TT _{faulty_IASEN-2}	TT of IASEN-2 of network size N in faulty condition
TT _{faulty_IAON}	TT of IAON of network size N in faulty condition
TT _{faulty_IASEN-3}	TT of IASEN-3 of network size N in faulty condition
TP _{MALN}	Throughput of MALN in non-faulty condition
TP _{IASEN-2}	Throughput of IASEN-2 in non-faulty condition
TP _{IAON}	Throughput of IAON in non-faulty condition
TP _{IASEN-3}	Throughput of IASEN-3 in non-faulty condition
TP _{faulty_MALN}	Throughput of MALN in faulty condition
TP _{faulty_IASEN-2}	Throughput of IASEN-2 in faulty condition
TP _{faulty_IAON}	Throughput of IAON in faulty condition
TP _{faulty_IASEN-3}	Throughput of IASEN-3 in faulty condition

Table 4.1.Symbol Table.

Symbol	Meaning of Symbol
PU_{MALN}	Processor utilization of MALN in non-faulty condition
$PU_{IASEN-2}$	Processor utilization of IASEN-2 in non-faulty condition
PU_{IAON}	Processor utilization of IAON in non-faulty condition
$PU_{IASEN-3}$	Processor utilization of IASEN-3 in non-faulty condition
PU_{faulty_MALN}	Processor utilization of MALN in faulty condition
$PU_{faulty_IASEN-2}$	Processor utilization of IASEN-2 in faulty condition
PU_{faulty_IAON}	Processor utilization of IAON in faulty condition
$PU_{faulty_IASEN-3}$	Processor utilization of IASEN-3 in faulty condition
PP_{MALN}	Processor power of MALN in non-faulty condition
$PP_{IASEN-2}$	Processor power of IASEN-2 in non-faulty condition
PP_{IAON}	Processor power of IAON in non-faulty condition
$PP_{IASEN-3}$	Processor power of IASEN-3 in non-faulty condition
PP_{faulty_MALN}	Processor power of MALN in faulty condition
$PP_{faulty_IASEN-2}$	Processor power of IASEN-2 in faulty condition
PP_{faulty_IAON}	Processor power of IAON in faulty condition
$PP_{faulty_IASEN-3}$	Processor power of IASEN-3 in faulty condition
SE-a	Switching element a of the network
SE-b	Switching element b of the network
SE-c	Switching element c of the network
SE-d	Switching element d of the network
SE-e	Switching element e of the network
SE-f	Switching element f of the network
SE-g	Switching element g of the network
SE-h	Switching element h of the network
SE-i	Switching element i of the network
SE-j	Switching element j of the network
SE-k	Switching element k of the network
SE-l	Switching element l of the network
SE-m	Switching element m of the network
SE-n	Switching element n of the network
SE-o	Switching element o of the network
SE-p	Switching element p of the network
SE-q	Switching element q of the network
SE-r	Switching element r of the network

Table 4.2.Symbol Table.

4.3 Irregular Advance Omega Network

This section gives a detail description of IAON.

4.3.1 Structure of Irregular Advance Omega Network

Irregular advance omega network is derived from advance omega network [16]. In figure 4.1, the source address, destination address, multiplexer and demultiplexer are represented by S, D, Mux and Demux respectively.

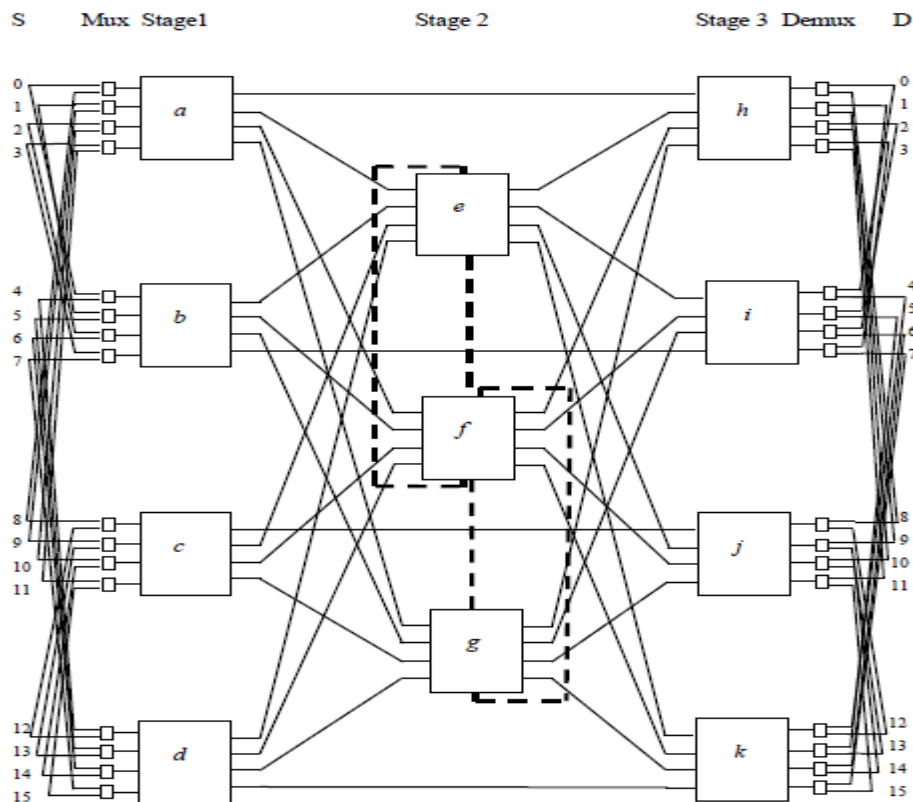


Figure 4.1.16x16 IAON.

In IAON, every source has connectivity with two SEs of first stage through auxiliary links. Similarly, every destination has connectivity with two SEs of third stage through auxiliary links. The auxiliary links of second stage are shown by dotted lines in figure 4.1. The network has 1 primary and 2 alternate path between every source and destination. If a SE has direct connectivity with any source or destination then it will be known as primary SE. If a SE has connectivity with any source or destination through auxiliary links then it will be named as first and second alternate SE respectively.

Based on this assumption, it is understood that SE a, b and c will be the primary, first alternate and second alternate SEs for source 0, respectively. Similarly, the primary, first alternate, and second alternate SEs for other source or destination can be obtained. Figure 4.2 shows the redundancy graph of IAON. In this graph, the source and destination are shown by empty circle. In figure 4.2, all the SEs are depicted by black nodes. Redundancy graph tells that the IAON has the potency of good communication in faulty and non-faulty situations.

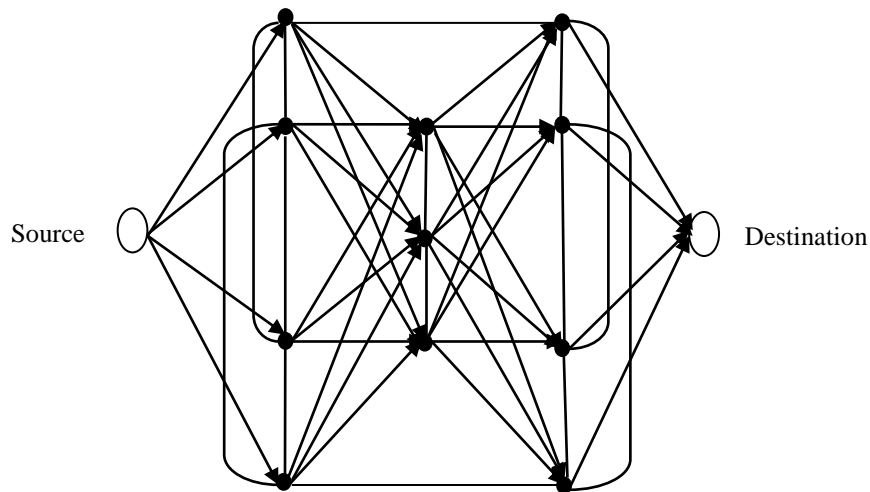


Figure 4.2.Redundancy Graph of IAON.

4.3.2 Routing Algorithm of IAON

The routing algorithm of IAON shows that how the data packets can be routed through the network in non-faulty, single switch faulty, and double switch faulty conditions. As an input, initially the user will enter the size of network (N) and as an output he will get the status that the transmitted data packets reached on N destinations or not reached. The routing algorithm of IAON is shown in figure 4.3

Algorithm_IAON_Broadcast

Input: N

Output: Data Packets Successfully Reached on N Destinations or Network Fails

BEGIN

1. for i = 0 to N
2. Source = i
3. FIRST-STAGE (Source)

FIRST-STAGE (Source)

1. if $SE_{P_1} == F$
2. First_Alternate_ SE_1
3. else SECOND-STAGE (Source)
4. if First_Alternate_ $SE_1 == F$
5. Second_Alternate_ SE_1
6. else SECOND-STAGE (Source)
7. if Second_Alternate_ $SE_1 == F$
8. Network Fails
9. else SECOND-STAGE (Source)

SECOND-STAGE (Source)

1. if SE-e == F
2. SE-f
3. else THIRD-STAGE (Source)
4. if SE-f == F
5. SE-g
6. else THIRD-STAGE (Source)
7. if SE-g == F
8. Network Fails
9. else THIRD-STAGE (Source)

THIRD-STAGE (Source)

1. if $SE_{P_3} == F$
2. First_Alternate_ SE_3
3. else Collect data packets on SE_{P_3} and Send to N destinations
4. if First_Alternate_ $SE_3 == F$
5. Second_Alternate_ SE_3
6. else Collect data packets on First_Alternate_ SE_3 and Send to N destinations
7. if Second_Alternate_ $SE_3 == F$
8. Network Fails
9. else Collect data packets on Second_Alternate_ SE_3 and Send to N destinations

END

Figure 4.3. Routing Algorithm of IAON.

This network has 3 functions which are as follows:

- FIRST-STAGE (Source)
- SECOND-STAGE (Source)

- THIRD-STAGE (Source)

The main functionality of all these functions is to provide a suitable path in such a way that in faulty or non-faulty conditions data packets can be reached from a given source to all N destinations. If it is found that 1 SE or 2 SEs of a stage are faulty then data packets will arrive on another SE of that particular stage. The SE which collects the data packets will send them towards the N destinations. This is the complete routing process of IAON.

4.3.2.1 Proof of Correctness

To prove the algorithm the author has taken an example which is as follows:

Example 1: It is assumed that the source is 5 and data packets have to be sent to N destinations. In this example, the author has considered three cases which are as follows:

Case1: When SEs are not faulty in every stage.

If the network is non-faulty then the primary routes of source 5 will be as follows:

Route 1: *b-e-h*,

Route 2: *b-e-i*,

Route 3: *b-e-j*,

Route 4: *b-e-k*.

After reaching on SE-h, SE-i, SE-j and SE-k data packets will be transmitted to all the given destinations through Demux. The routes are shown by dotted lines in figure 4.4.

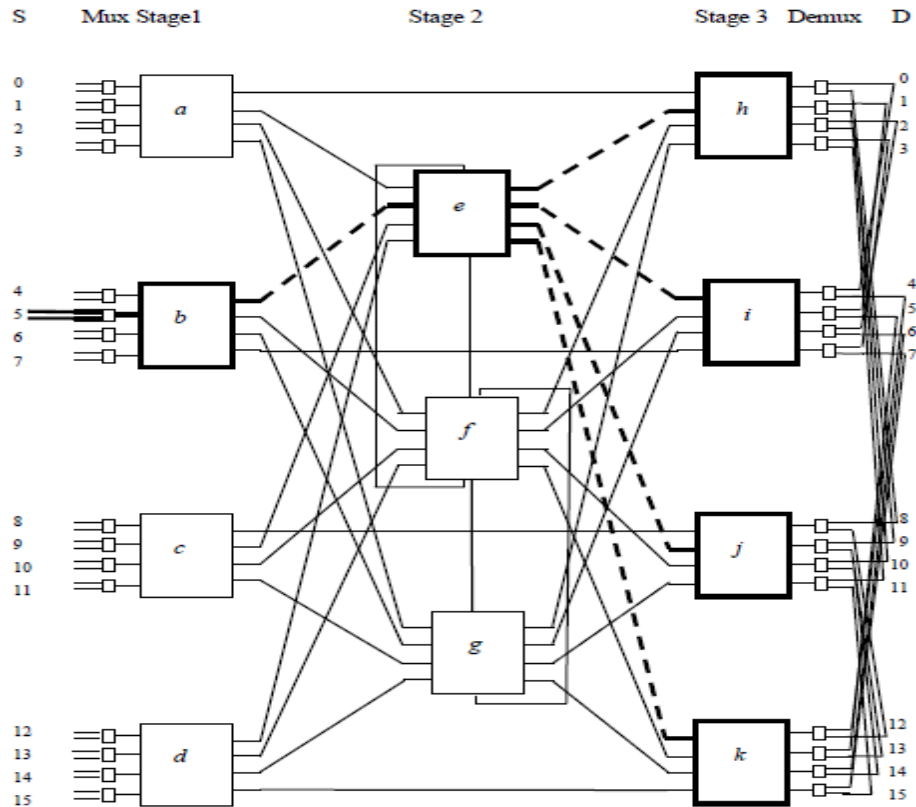


Figure 4.4. Primary Routes of Source 5 in IAON.

Case 2: When 1 SE of each stage is faulty.

Here it is assumed that SEs b, e and h are faulty. In this case the first alternate routes of source 5 will be as follows:

- Route 1: *a-f-i*,
- Route 2: *a-f-j*,
- Route 3: *a-f-k*.

In figure 4.5, it can be seen that SE-b has connectivity with SE-a through Mux. Therefore, SE-a will be the first alternate SE of stage 1 for source 5. In the same way, SE-f will be the first alternate SE of stage 2 for source 5. In figure 4.5, it can be seen that SE-h has connectivity with SE-j and SE-j. Hence, SE-i and SE-j will become the first alternate SEs of stage 2. The routes are shown by dotted lines in figure 4.5.

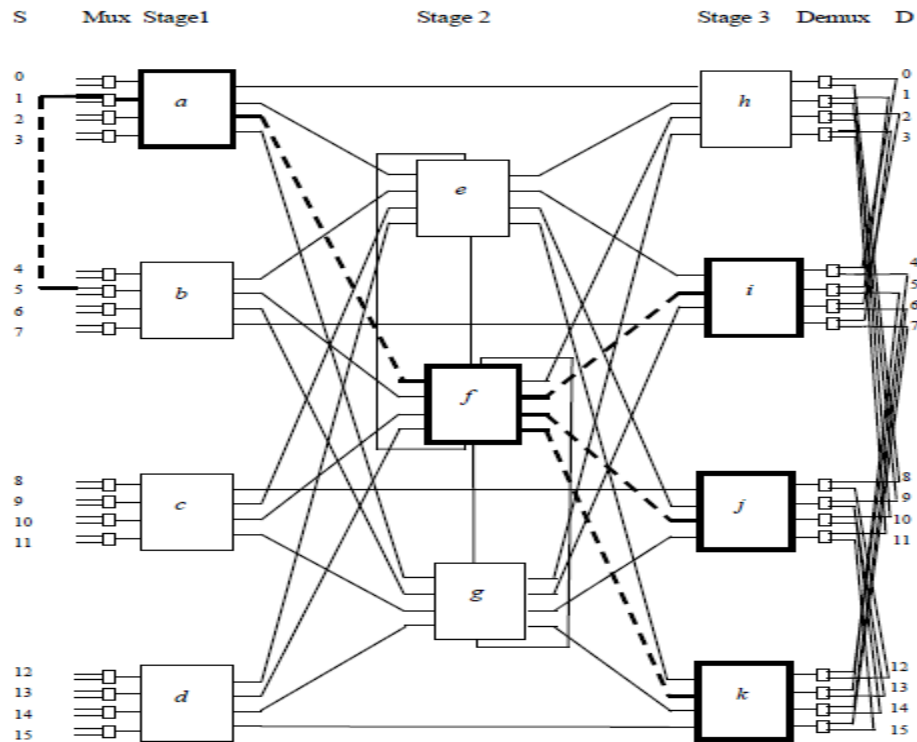


Figure 4.5. First Alternate Routes of Source 5 in IAON.

Case 3: When 2 SEs of each stage are faulty.

Here it is assumed that SEs b, a, e, f, h and i are faulty. In this case the second alternate routes of source 5 will be as follows:

- Route 1: $d-g-j$,
- Route 2: $d-g-k$.

In figure 4.6, it can be seen that SE-b has connectivity with SE-d through a Mux. Therefore, SE-d will be the second alternate SE of stage 1 for source 5. In the same way, SE-g will be the second alternate SE of stage 2 for source 5. In figure 4.6, it can be seen that SE-h and SE-i have connected with SE-j and SE-k. Hence, SE-j and SE-k will become the second alternate SEs of stage 2. The routes are shown by dotted lines in figure 4.6.

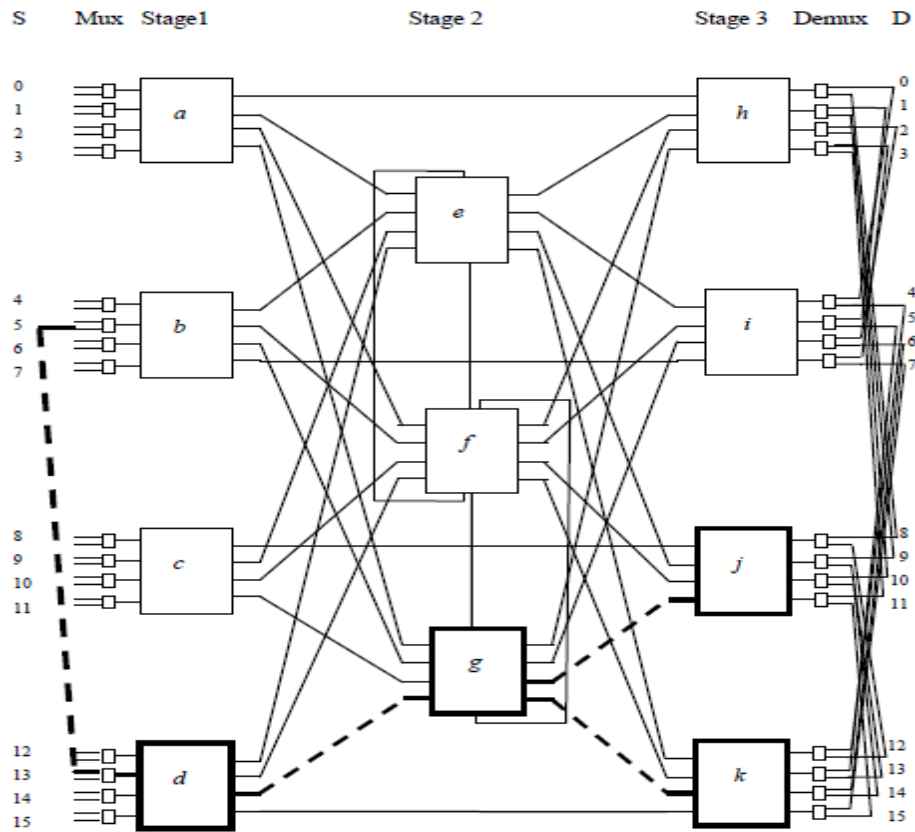


Figure 4.6. Second Alternate Routes of Source 5 in IAON.

Here the author has discussed three cases. This discussion shows that the proposed MIN is a double fault tolerant interconnection network means it can sustain 2 faulty SEs in each stage simultaneously. If the network has more than 2 faulty SE in each stage simultaneously, then the network will fail in order to provide the alternate routes.

Theorem: There are at least 21 routes between every source and destination address.

Proof: In the previous subsections, it is shown in all figures of IAON that the second stage of IAON unites the SEs of first stage and third stage. It means that the second stage is a very important stage of IAON. The other point is that all the alternate routes are generated because of second stage. In IAON, when data packets come from any SE of first stage via any source, then definitely they will go on any one SE of second stage. In this case, the routes via SEs of the second stage will be as follows:

Route 1: SE1-*e*-SE3,
Route 2: SE1-*f*-SE3,
Route 3: SE1-*g*-SE3,
Route 4: SE1-*e-f*-SE3,
Route 5: SE1-*f-e*-SE3,
Route 6: SE1-*f-g*-SE3,
Route 7: SE1-*g-f*-SE3.

Furthermore, it is already discussed that every source is allied with 3 SEs of stage 1. Therefore, total available routes will be

$$\text{IAON}_{\text{Route}} = (7 \times \text{Allied SE1}),$$

$$\text{IAON}_{\text{Route}} = (7 \times 3),$$

$$\text{IAON}_{\text{Route}} = 21.$$

$\text{IAON}_{\text{Route}}$ represents the total number of routes in IAON. Therefore, it is proved that the network has atleast 21 routes between every source and destination. To understand the theorem appropriately the author has taken an example.

Example 2: It is assumed that the source is 9 and the destination is 2. In figure 4.1, it can be observed that source 9 has connectivity with SEs *c*, *a* and *d*. Further, a destination 2 has connectivity with SEs *h*, *i* and *j*. Therefore, the possible alternate routes will be as follows:

Route 1: *c-e-h*,
Route 2: *c-f-h*,
Route 3: *c-g-h*,
Route 4: *c-e-f-h*,
Route 5: *c-f-e-h*,
Route 6: *c-f-g-h*,
Route 7: *c-g-f-h*,
Route8: *a-e-h*,

Route9: *a-f-h*,
Route10: *a-g-h*,
Route11: *a-e-f-h*,
Route12: *a-f-e-h*,
Route13: *a-f-g-h*,
Route14: *a-g-f-h*,
Route 15: *d-e-h*,
Route16: *d-f-h*,
Route17: *d-g-h*,
Route18: *d-e-f-h*,
Route19: *d-f-e-h*,
Route20: *d-f-g-h*,
Route21: *d-g-f-h*.

Therefore, again, it is proved that there are atleast 21 routes between every source and destination address.

4.4 Irregular Augmented Shuffle Exchange Network-3

This section gives a detail description of IAON.

4.4.1 Structure of Irregular Augmented Shuffle Exchange Network-3

The structure of irregular augmented shuffle exchange network-3 is based on IASEN-2. In figure 4.7, It can be seen that it has 16 sources and 16 destinations, hence the size of IASEN-3 is $N = 16$.

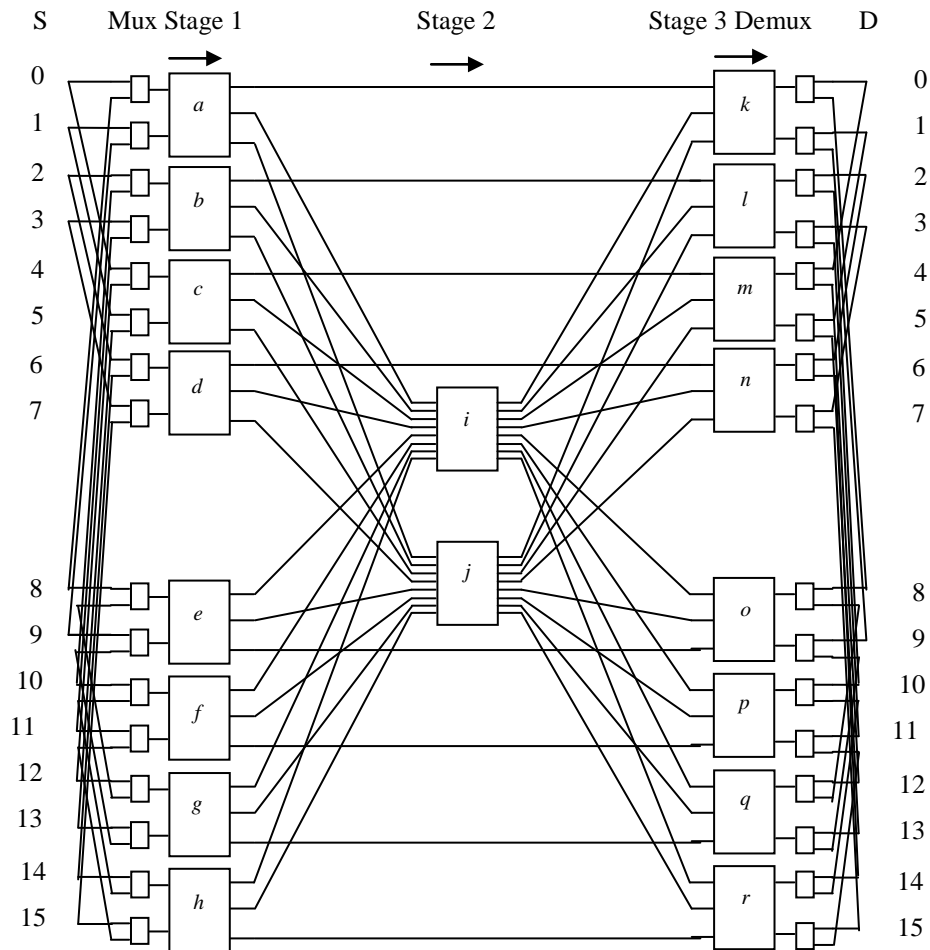


Figure 4.7. 16×16 IASSEN-3.

In figure 4.7, the source address, destination address, multiplexer and demultiplexer are represented by S, D, Mux and Demux respectively. This is a 3-stage MIN. In first and last stage, each source or each destination is connected with three SEs of that particular stage, e.g. source 11 is connected with SE *f*, *a* and *d* and therefore *f*, *a* and *d* are the primary, first alternate and second alternate SEs for source 11. In the same way, the primary, first alternate and second alternate SEs for another source and destinations can be obtained. The size of each SE in first and third stage is 2×3 and 3×2 respectively. In stage 2, the size of each SE is 8×8.

4.4.2 Routing Algorithm of IASSEN-3

The routing algorithm of IASSEN-3 is given in figure 4.8.

Algorithm_IASEN-3

Input: N, Source, Destinations

Output: Data Packets Reached Successfully or Network Fails

BEGIN

1. if $PSE_1 == F \parallel PSE_3 == F$
2. then AE1
3. else Send data packets to Next Appropriate Node
4. if AE1 == F
5. then AE2
6. else Send data packets to Next Appropriate Node
7. if AE2 == F
8. then Network Fails
9. if SE-i == F
10. then SE-j
11. else Send data packets to Appropriate SE of Third Stage
12. if SE-j == F
13. then Network Fails
14. else Send data packets to Appropriate SE of Third Stage

End

Figure 4.8. Routing algorithm of IASEN-3.

In the routing algorithm of IASEN-3, if data packets arrive at the primary SE of the first stage (PSE1) or primary SE of third stage (PSE3), then that particular SE will be checked that whether it is faulty or non-faulty. If it is noticed that the required SE is faulty then the first alternate SE (AE1) of the first stage will get the data packets. If AE1 is also faulty then second alternate SE (AE2) of the first stage will get the data packets. In case all the SEs i.e. PSE1, AE1 and AE2 of first stage or PSE3, AE1 and AE2 of third stage are not responding properly then the network will fail. If data packets arrive at SE i of second stage and it is faulty then SE j of the second stage will receive the request. If it is noticed that all the required SEs of a stage are faulty then the network will fail. Finally, it can be said that, if the required SEs of all three stages are in working condition, then data will be transmitted from the given source to its given destinations otherwise the data transmis-

sion process will be stopped. In the third step of the algorithm the term “Node” may be the SE of second stage or the given destination.

Theorem: IASEN-3 shows single switch fault tolerability in each stage simultaneously.

Proof: To prove the theorem the author has taken an example which is as follows:

Example 3: It is assumed that the source is 2 and the destination is 9 and SEs b, i, and o are faulty. In this situation the rest of the routes are as follows:

Route 1: 2-6-Mux-d-j-k-Demux-1-9

Route 2: 2-10-Mux-f-j-k-Demux-1-9

Route 3: 2-6-Mux-d-j-q-Demux-13-9

Route 4: 2-10-Mux-f-j-q-Demux-13-9

These available paths prove that IASEN-3 is a single switch fault tolerant network in every stage.

4.5 Performance Analysis and Simulation Results

In this section, the author has presented all the results which he has got in the simulations. The performance of MALN [21], IASEN-2 and IAON are obtained in non-faulty and single switch faulty conditions. It is observed that IAON has better performance than the MALN and IASEN-2 in both conditions. The reasons are as follows:

- i) IAON has better bandwidth than the MALN and IASEN-2.
- ii) If a network has good bandwidth means it will have good throughput, good processor utilization and good processing power. Therefore IAON is better than the MALN and IASEN-2.
- iii) It has better probability of acceptance (PA) than the MALN and IASEN-2. PA of IAON is better means the packet loss probability in IAON is less than the MALN and IASEN-2.
- iv) Further, IAON is double switch fault tolerant MIN means it can sustain 2 faulty SEs in stage simultaneously whereas MALN and IASEN-2 are single switch fault tolerant MIN means these MINs can sustain single faulty SE in each stage simultaneously.

Further, the author has compared the performance of IASEN-2, IAON and IASEN-3 in non-faulty and single switch faulty conditions. It is observed that IASEN-3 has better performance than the IASEN-2 and IAON in both conditions. IASEN-3 has the same reasons of better performance as the IAON has, except the last (4th) reason. IASEN-3 is a single switch fault tolerant network. The author has also shown the performance of IAON in non-faulty, single switch faulty and double switch faulty conditions. The author has considered the bandwidth, the probability of acceptance, transmission time, throughput, processor utilization, and processing power as the factors of performance. The descriptions of all these factors are given in the next subsections.

4.5.1 Load Factor

The term load factor or request generation probability means that how much load is offered to the network for the data transmission. Here, the load factor has 10 different values start from 0.1 and goes up to 1 by adding 0.1 in the previous value [16, 21, 22]. Different numbers of data packets are transmitted from the given source to all destinations on each value of load factor. In this simulation the author has considered the following table:

Load Factor (p)	Total Number of Data packets
0.1	50
0.2	100
0.3	150
0.4	200
0.5	250
0.6	300
0.7	350
0.8	400
0.9	450
1.0	500

Table 4.3. Load on Given Load Factors.

According to table 4.3, the author has transmitted the data packets through MALN, IASEN-2, IAON, and IASEN-3 in faulty and non-faulty conditions on each value of load factor. Here the term “faulty” means single switch faulty for MALN, IASEN-2, and IASEN-3. For IAON, “faulty” means single and double switch faulty conditions. The size of each data packet is 1 byte in the simulation.

4.5.2 Calculation of Bandwidth

To calculate the bandwidth (BW), the author has applied the probabilistic method on MALN [21], IASEN-2, IAON and IASEN-3. To understand the probabilistic method, he has given an example. It is assumed that a SE k has x number of input lines and y number of output lines and therefore the size of the SE is $x \times y$. When data packets arrive on SE k then it collects them and forward to other SE or destination. Hence, according to probabilistic method [16]:

- If the data packets are not arriving on SE k then the probability will be $\{1 - (p/y)\}$. Here p is the load factor or probability of data packets which are generated by a source.
- If the data packets are not arriving on SE k from “ x ” input lines, then the probability will be $\{1 - (p/y)\}^x$.
- If the data packets are arriving on one output line of SE k from “ x ” input lines of SE k then the probability will be $1 - \{1 - (p/y)\}^x$.

Therefore, SE k passes the total number of data packets per unit time will be $y - \{1 - (p/y)\}^x$. A MIN consists of more than one stage and each stage has a certain number of SEs. Hence, the output rate of a stage will become the input rate of the next stage. Based on this phenomenon the author has calculated the BW of MALN, IASEN-2, IAON, and IASEN-3. Basically, bandwidth is the capacity of a network. It can be defined as follows:

Definition 1: “Bandwidth is the maximum rate at which information can be transferred [5, 16, 21, 22]”.

Based on the probabilistic method, BW of a MIN will be = (total number of output lines \times p) bytes/second.

Hence, probability equations of MALN [21]:

$$p_1 = 1 - \left(1 - \frac{p_0}{3}\right)^3$$

$$p_2 = 1 - \left(1 - \frac{p_1}{6}\right)^3$$

$$p_3 = 1 - \left(1 - \frac{p_2}{3}\right)^3$$

$$p_4 = 1 - \left\{ \left(1 - p_3\right) \times \left(1 - \frac{p_1}{2}\right) \right\}^2$$

$$BW_{MALN} = (N \times p_4)$$

Here p_1 , p_2 , p_3 and p_4 represents the load (data packets) arrived on the SEs of first, second, third, and fourth stage of MALN.

Probability equations of IASEN-2:

$$p_1 = 1 - \left(1 - \frac{p_0}{3}\right)^2$$

$$p_2 = 1 - \left(1 - \frac{p_1}{6}\right)^9$$

$$p_3 = 1 - \left(1 - \frac{p_2}{9}\right)^3$$

$$p_4 = 1 - \left\{ \left(1 - p_3\right) \times \left(1 - \frac{p_1}{2}\right) \right\}^3$$

$$BW_{IASEN-2} = (N \times p_4)$$

Here p_1 , p_2 , p_3 and p_4 represents the load (data packets) arrived on the SEs of first, second, third and fourth stage of IASEN-2.

Probability equations of IAON:

$$p_1 = 1 - \left(1 - \frac{p_0}{4}\right)^4$$

$$p_2 = 1 - \left(1 - \frac{p_1}{16}\right)^5$$

$$p_3 = 1 - \left\{ \left(1 - p_2\right) \times \left(1 - \frac{p_1}{4}\right) \right\}^4$$

$$BW_{IAON} = (N \times p_3)$$

Here p_1 , p_2 , and p_3 represents the load (data packets) arrived on the SEs of first, second and third stage of IAON.

Probability equations of IASEN-3:

$$p_1 = 1 - \left(1 - \frac{p_0}{3}\right)^2$$

$$p_2 = 1 - \left(1 - \frac{p_1}{8}\right)^8$$

$$p_3 = 1 - \left\{ \left(1 - p_2\right) \times \left(1 - \frac{p_1}{2}\right) \right\}^3$$

$$BW_{IASEN-3} = (N \times p_3)$$

Here p_1 , p_2 , and p_3 represents the load (data packets) arrived on the SEs of first, second and third stage of IASEN-3.

4.5.3 Calculation of Probability of Acceptance

It is not certain that the total number of data packets which are transmitted from a source and the total number of data packets which are accepted by N destinations will be the same. Loss of data packets occurs during the data transmission process because of switch failure, link failure, or any other reason. Therefore, the probability of acceptance (PA) gives the information that how many data packets will be accepted by N destinations. It can be defined as follows [5, 21]:

Definition 2: "Probability of Acceptance is defined as the ratio of the expected bandwidth to the expected number of requests generated per cycle [5, 21, 22]".

Therefore, PA of MALN, IASEN-2, IAON and IASEN-3 will be as follows:

$$PA_{MALN} = (BW_{MALN}/N \times p)$$

$$PA_{IASEN-2} = (BW_{IASEN-2}/N \times p)$$

$$PA_{IAON} = (BW_{IAON}/N \times p)$$

$$PA_{IASEN-3} = (BW_{IASEN-3}/N \times p)$$

4.5.4 Calculation of Transmission Time

The transmission time is the time which is taken by the data packets from any source to any given destination [1-6]. To calculate the transmission time the author has taken some assumptions.

- To understand the transmission time, the author has taken a sample network, which is shown in figure 4.9.

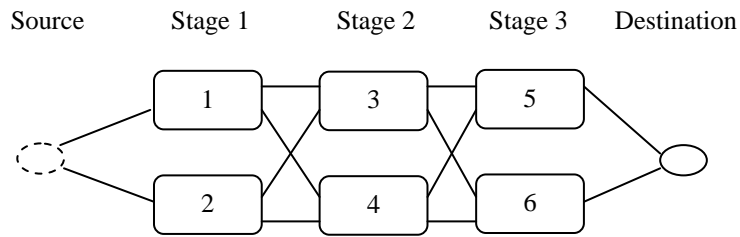


Figure 4.9. Sample Network.

In figure 4.9, source and destination node is shown by dotted circle and a solid circle respectively. It has three stages therefore; a data packet can take any one route:

- Route1: Source - SE1 - SE3 - SE5 - Destination,
- Route2: Source - SE1 - SE4 - SE5 - Destination,
- Route3: Source - SE1 - SE3 - SE6 - Destination,
- Route4: Source - SE1 - SE4 - SE6 - Destination,
- Route5: Source - SE2 - SE3 - SE5 - Destination,
- Route6: Source - SE2 - SE4 - SE5 - Destination,
- Route7: Source - SE2 - SE3 - SE6 - Destination,
- Route8: Source - SE2 - SE4 - SE6 - Destination.

It is assumed that a data packet takes 1 second from one node to another node in non-faulty condition. Here the nodes may be any SE or any processor. Therefore, the transmission time of a data packet will be 4 seconds. It means if the number of stages is 3 then transmission time of a data packet is $3+1=4$ seconds.

Hence, if the number of stages in a MIN is q then transmission time of a data packet will be:

$$TT_1 = (q + 1)$$

Here TT_1 is the transmission time of a single data packet. If there are i numbers of data packets which are to be sent from a given source to N destinations then transmission time will be:

$$TT_i = (q + i)$$

Here TT_i is the transmission time of i number of data packets. Calculation of TT_i can be understood by figure 4.9.

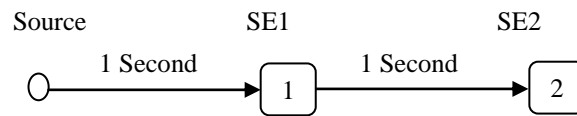


Figure 4.10. Calculation of Data Transmission Time.

In figure 4.10, when a data packet will move from the SE1 to the SE2 of the next stage, then in the mean time the second data packet will take the route from source to SE1. Therefore, if the size of the network is 16, then transmission time will be as follows:

$$TT_{i_{16}} = (q + i)$$

$TT_{i_{16}}$ is the transmission time of a network of size 16.

- If the size of the network (N) will increase then the number of SEs in each stage will also increase. Therefore, the TT_i will increase. In this chapter, the minimum network size is $N=16$ for each network. The transmission time for a large network will be calculated as follows:

$$TT_{i_N} = \{TT_{i_{16}} + \left(\frac{N}{16} - 1\right)\}$$

Here, TT_{i_N} is the transmission time of a network of size N .

- If the network is faulty then data packets will take extra transmission time. It is assumed that if k SEs are faulty then extra k seconds will be taken by the data packets. In this case the total transmission time will be as follows:

$$TT_{\text{faulty}} = TT_{i_N} + k$$

Here, TT_{faulty} is the transmission time of a network in faulty condition.

Therefore, the transmission time of MALN, IASEN-2, IAON and IASEN-3 in non-faulty case will be as follows:

$$TT_{i_N_{\text{MALN}}} = \{TT_{i_{16_{\text{MALN}}} + \left(\frac{N}{16} - 1\right)\}$$

$$TT_{i_N_{\text{IASEN-2}}} = \{TT_{i_{16_{\text{IASEN-2}}} + \left(\frac{N}{16} - 1\right)\}$$

$$TT_{i_N_IAON} = \{TT_{i_16_IAON} + \left(\frac{N}{16} - 1\right)\}$$

$$TT_{i_N_IASEN-3} = \{TT_{i_16_IASEN-3} + \left(\frac{N}{16} - 1\right)\}$$

Therefore, the transmission time of MALN, IASEN-2, IAON and IASEN-3 in the faulty case will be as follows:

$$TT_{\text{faulty_MALN}} = (TT_{i_N_MALN} + k)$$

$$TT_{\text{faulty_IASEN-2}} = (TT_{i_N_IASEN-2} + k)$$

$$TT_{\text{faulty_IAON}} = (TT_{i_N_IAON} + k)$$

$$TT_{\text{faulty_IASEN-3}} = (TT_{i_N_IASEN-3} + k)$$

In this way the author has calculated the transmission time of data packets which are transmitted through the MALN, IASEN-2, IAON and IASEN-3.

4.5.5 Calculation of Throughput

In a network, throughput (TP) means the average number of data packets accepted by the given network. It depends on the bandwidth and transmission time of the network. It can be defined as follows [5, 21, 22]:

Definition 3: "Throughput means the average number of cells delivered by a network per unit time [1-6, 21,22]".

Therefore, the TP of MALN, IASEN-2, IAON and IASEN-3 in non-faulty conditions are as follows [5, 21, 22]:

$$TP_{\text{MALN}} = \left(\frac{BW_{\text{MALN}}}{N \times TT_{i_N_MALN}}\right)$$

$$TP_{\text{IASEN-2}} = \left(\frac{BW_{\text{IASEN-2}}}{N \times TT_{i_N_IASEN-2}}\right)$$

$$TP_{\text{IAON}} = \left(\frac{BW_{\text{IAON}}}{N \times TT_{i_N_IAON}}\right)$$

$$TP_{\text{IASEN-3}} = \left(\frac{BW_{\text{IASEN-3}}}{N \times TT_{i_N_IASEN-3}}\right)$$

The TP of MALN, IASEN-2, IAON and IASEN-3 in faulty conditions are as follows:

$$TP_{\text{faulty_MALN}} = \left(\frac{BW_{\text{MALN}}}{N \times TT_{\text{faulty_MALN}}}\right)$$

$$TP_{\text{faulty_IASEN-2}} = \left(\frac{BW_{\text{IASEN-2}}}{N \times TT_{\text{faulty_IASEN-2}}} \right)$$

$$TP_{\text{faulty_IAON}} = \left(\frac{BW_{\text{IAON}}}{N \times TT_{\text{faulty_IAON}}} \right)$$

$$TP_{\text{faulty_IASEN-3}} = \left(\frac{BW_{\text{IASEN-3}}}{N \times TT_{\text{faulty_IASEN-3}}} \right)$$

In this way the author has calculated the throughput of MALN, IASEN-2, IAON and IASEN-3.

4.5.6 Calculation of Processor Utilization

Processor utilization (PU) means the time consumed by the processor in order to process the given workload. It depends on the given workload, the bandwidth of the network, and transmission time. Here the term “workload” refers the load factor. It can be defined as follows [1-6, 21, 22]:

Definition 4: “Processor Utilization is the expected percentage of time; a processor is active doing internal computation without accessing the global memory [5, 21, 22]”.

Therefore, the PU of MALN, IASEN-2, IAON and IASEN-3 in non-faulty conditions are as follows [5, 21, 22]:

$$PU_{\text{MALN}} = \left(\frac{TP_{\text{MALN}}}{p} \right)$$

$$PU_{\text{IASEN-2}} = \left(\frac{TP_{\text{IASEN-2}}}{p} \right)$$

$$PU_{\text{IAON}} = \left(\frac{TP_{\text{IAON}}}{p} \right)$$

$$PU_{\text{IASEN-3}} = \left(\frac{TP_{\text{IASEN-3}}}{p} \right)$$

The PU of MALN, IASEN-2, IAON and IASEN-3 in faulty conditions are as follows:

$$PU_{\text{faulty_MALN}} = \left(\frac{TP_{\text{faulty_MALN}}}{p} \right)$$

$$PU_{\text{faulty_IASEN-2}} = \left(\frac{TP_{\text{faulty_IASEN-2}}}{p} \right)$$

$$PU_{\text{faulty_IAON}} = \left(\frac{TP_{\text{faulty_IAON}}}{p} \right)$$

$$PU_{\text{faulty_IASEN-3}} = \left(\frac{TP_{\text{faulty_IASEN-3}}}{p} \right)$$

In this way the author has calculated the processor utilization of MALN, IASEN-2, IAON, and IASEN-3.

4.5.7 Calculation of Processing Power

The calculation of processing power (PP) is based on the processors which are used in order to complete the given workload. It depends on the processor utilization. It can be defined as follows [5, 21, 22]:

Definition 5: "Processing Power is defined as the sum of processor utilization over the number of processors [5, 21, 22]".

Therefore, the PP of MALN, IASEN-2, IAON and IASEN-3 in non-faulty conditions are as follows [5, 21, 22]:

$$PP_{\text{MALN}} = (PU_{\text{MALN}} \times N)$$

$$PP_{\text{IASEN-2}} = (PU_{\text{IASEN-2}} \times N)$$

$$PP_{\text{IAON}} = (PU_{\text{IAON}} \times N)$$

$$PP_{\text{IASEN-3}} = (PU_{\text{IASEN-3}} \times N)$$

Therefore, the PP of MALN, IASEN-2, IAON and IASEN-3 in faulty conditions are as follows:

$$PP_{\text{faulty_MALN}} = (PU_{\text{faulty_MALN}} \times N)$$

$$PP_{\text{faulty_IASEN-2}} = (PU_{\text{faulty_IASEN-2}} \times N)$$

$$PP_{\text{faulty_IAON}} = (PU_{\text{faulty_IAON}} \times N)$$

$$PP_{\text{faulty_IASEN-3}} = (PU_{\text{faulty_IASEN-3}} \times N)$$

In this way the author has calculated the processing power of MALN, IASEN-2, IAON and IASEN-3.

Note: In this chapter the author has taken the assumption that when the size of the network will increase then the number of SEs in each stage will increase. In his published work, he has taken the assumption that when the size of the network will increase then the size of SEs will increase.

4.5.8 Bandwidth for MALN, IASEN-2 and IAON

In figure 4.11, 4.12, 4.13 and 4.14, the author has shown the bandwidth of MALN, IASEN-2 and IAON for $N=16, 32, 64$ and 128 respectively. In all figures, it is clear that the IAON has better bandwidth than the earlier proposed MINs.

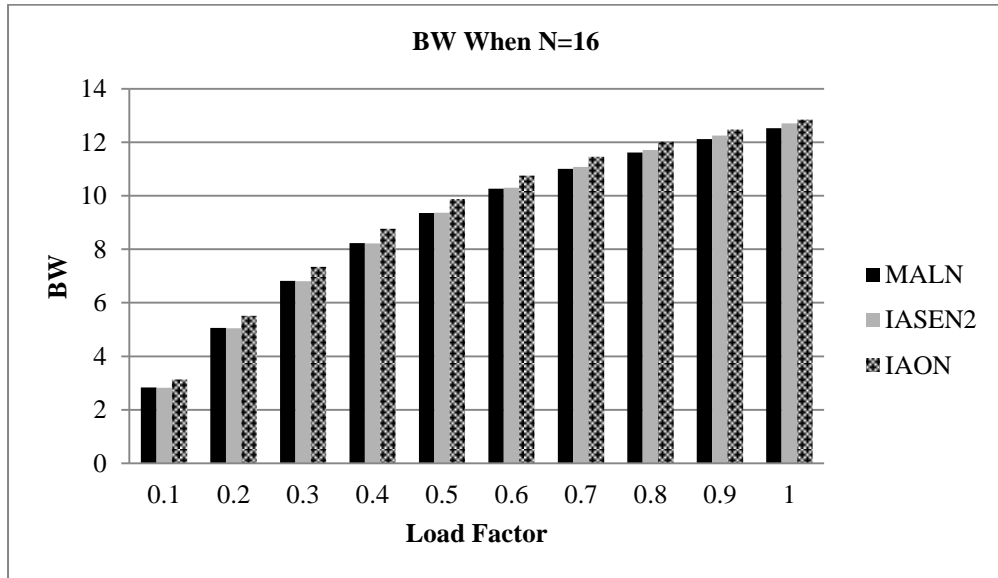


Figure 4.11. Comparison of BW for MALN, IASEN-2 and IAON when $N=16$.

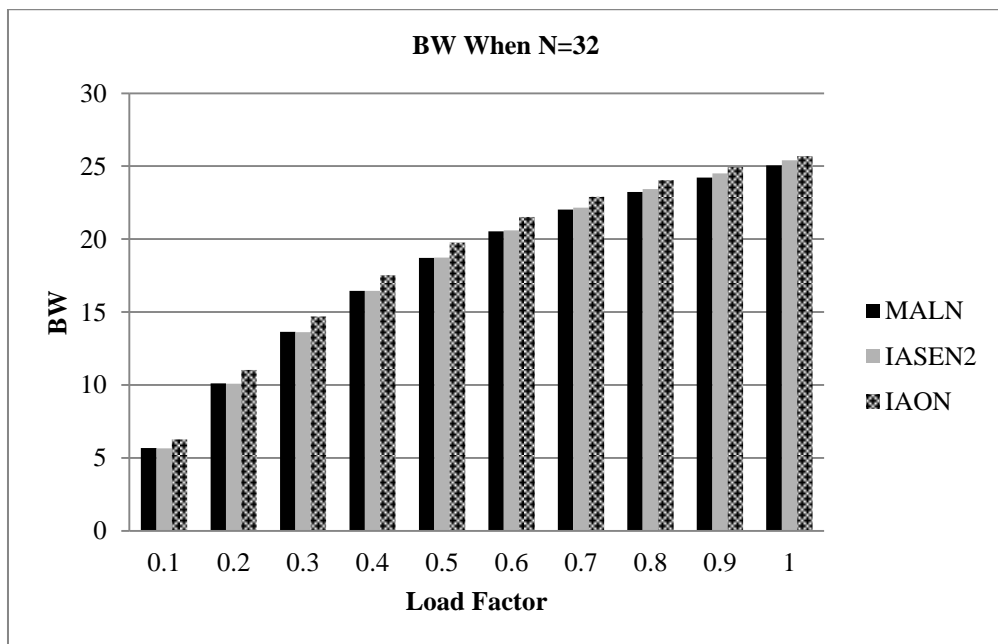


Figure 4.12. Comparison of BW for MALN, IASEN-2 and IAON when $N=32$.

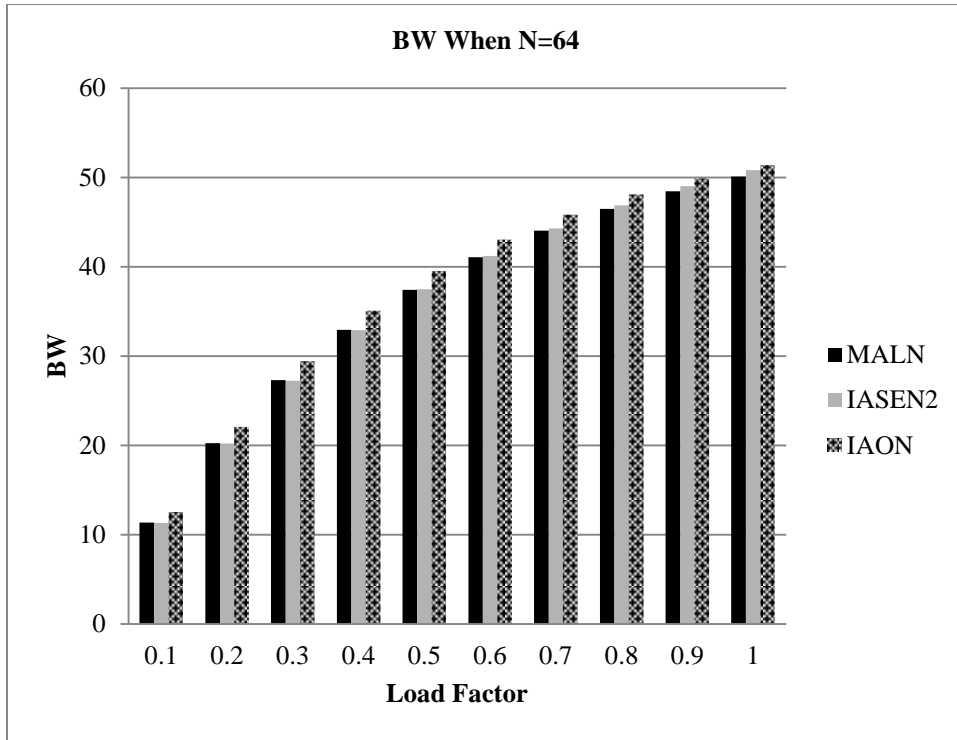


Figure 4.13. Comparison of BW for MALN, IASSEN-2 and IAON when N=64.

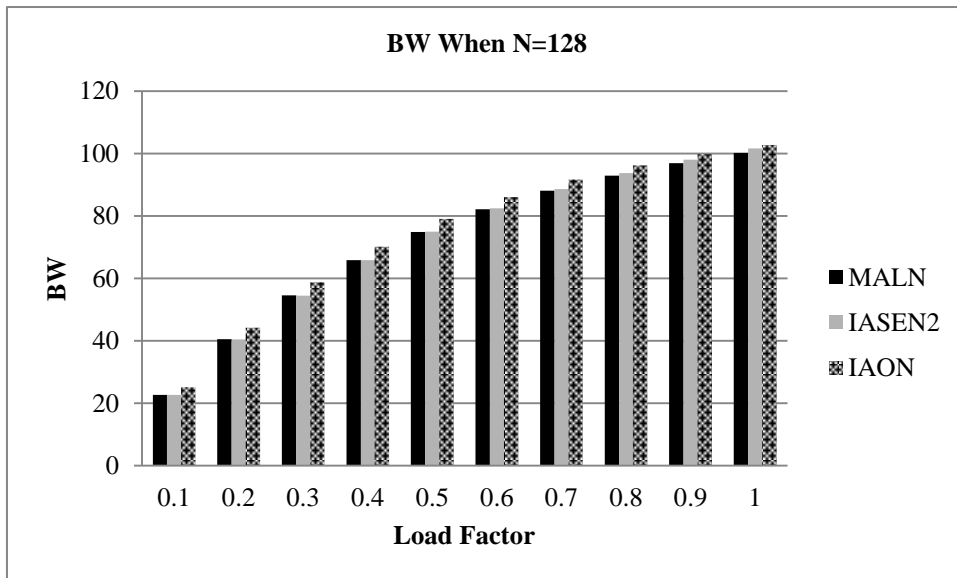


Figure 4.14. Comparison of BW for MALN, IASSEN-2 and IAON when N=128.

4.5.9 Probability of Acceptance for MALN, IASSEN-2 and IAON

In figure 4.15, the author has shown the bandwidth of MALN, IASSEN-2 and IAON. In this figure, it is clear that the IAON has a better probability of acceptance than the earlier proposed MINs.

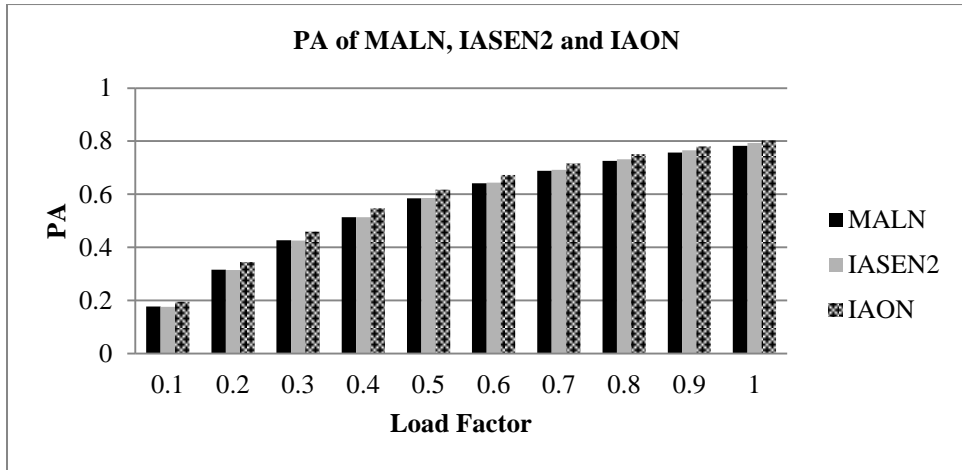


Figure 4.15. Comparison of PA for MALN, IASEN-2 and IAON.

4.5.10 Throughput for MALN, IASEN-2 and IAON in Non-Faulty Condition

In figure 4.16, 4.17, 4.18 and 4.19, the author has shown the throughput of MALN, IASEN-2 and IAON for N=16, 32, 64 and 128 respectively. In all figures, it is clear that the IAON has better throughput than the earlier proposed MINs. Here the throughput is calculated in non-faulty condition.

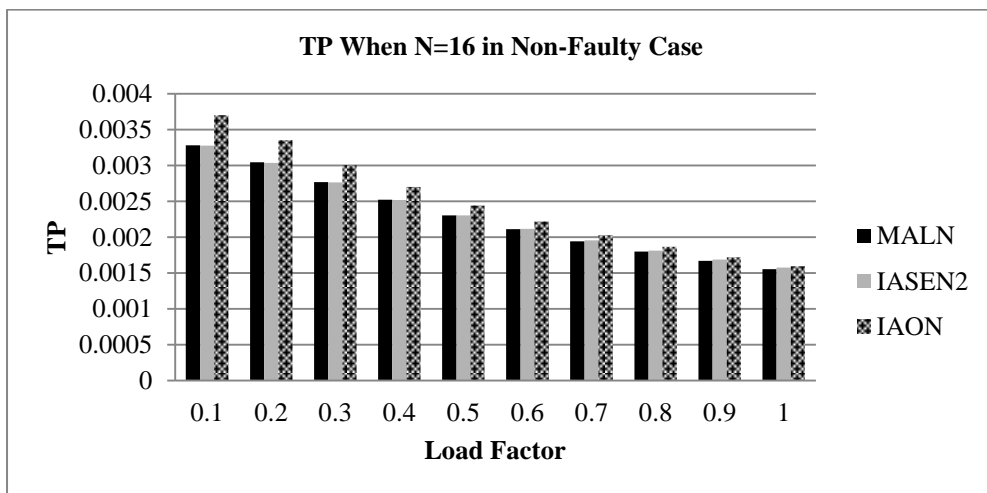


Figure 4.16. Comparison of TP for MALN, IASEN-2 and IAON when N=16 in Non-Faulty Condition.

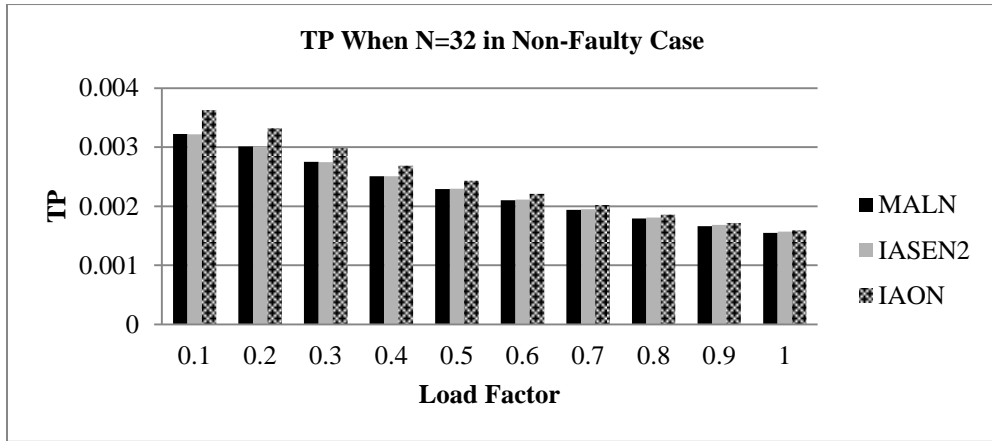


Figure 4.17. Comparison of TP for MALN, IASEN-2 and IAON when N=32 in Non-Faulty Condition.

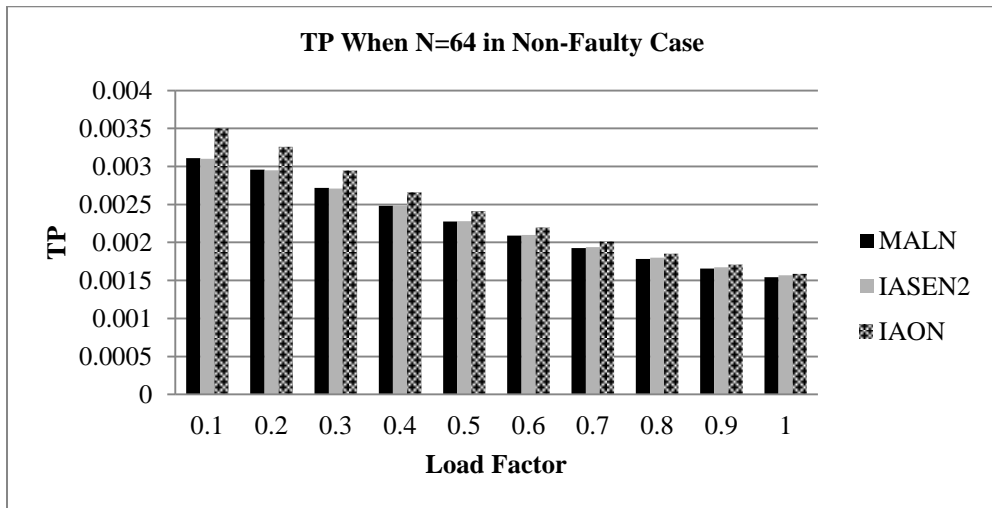


Figure 4.18. Comparison of TP for MALN, IASEN-2 and IAON when N=64 in Non-Faulty Condition.

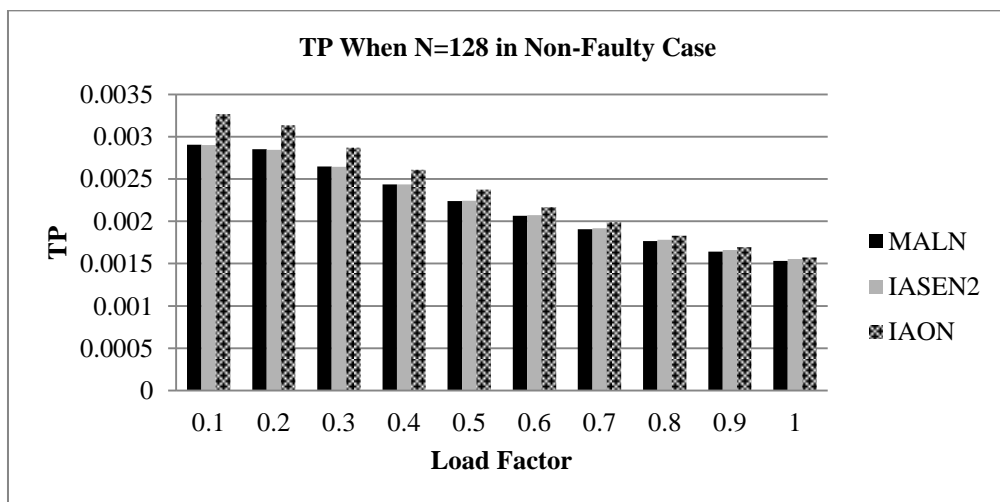


Figure 4.19. Comparison of TP for MALN, IASEN-2 and IAON when N=128 in Non-Faulty Condition.

4.5.11 Throughput for MALN, IASEN-2 and IAON in Single Switch Faulty Condition

In figure 4.20, 4.21, 4.22 and 4.23, the author has shown the throughput of MALN, IASEN-2 and IAON for $N=16, 32, 64$ and 128 respectively. Here the throughput is calculated in single switch faulty condition. In all figures, it is clear that the IAON has better throughput than the earlier proposed MINs.

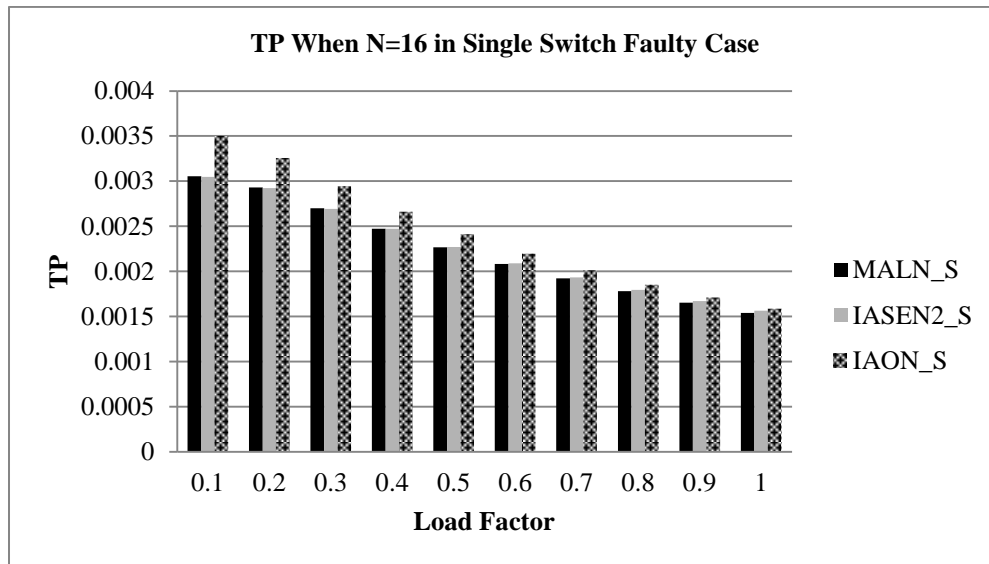


Figure 4.20. Comparison of TP for MALN, IASEN-2 and IAON when $N=16$ in Single Switch Faulty Condition.

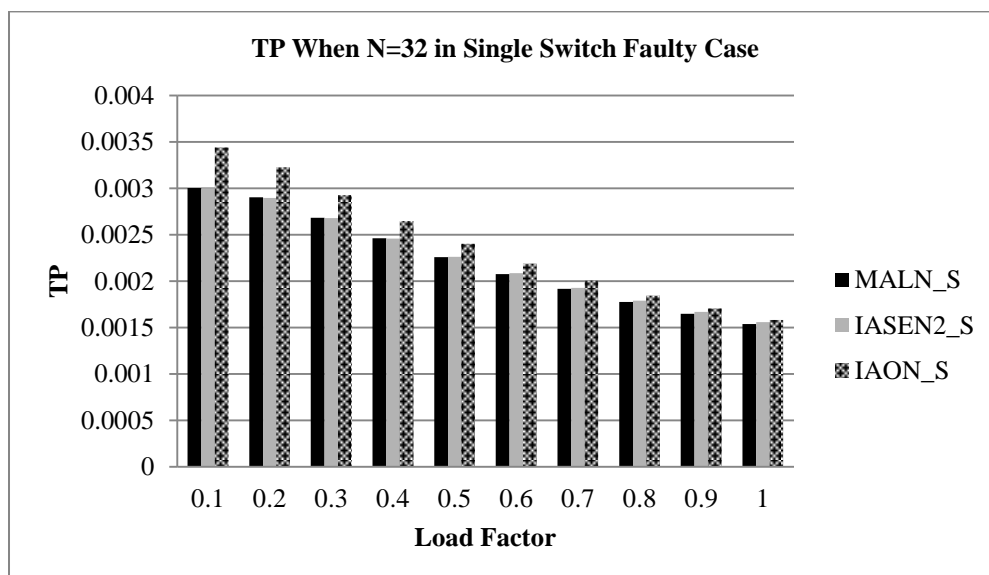


Figure 4.21. Comparison of TP for MALN, IASEN-2 and IAON when $N=32$ in Single Switch Faulty Condition.

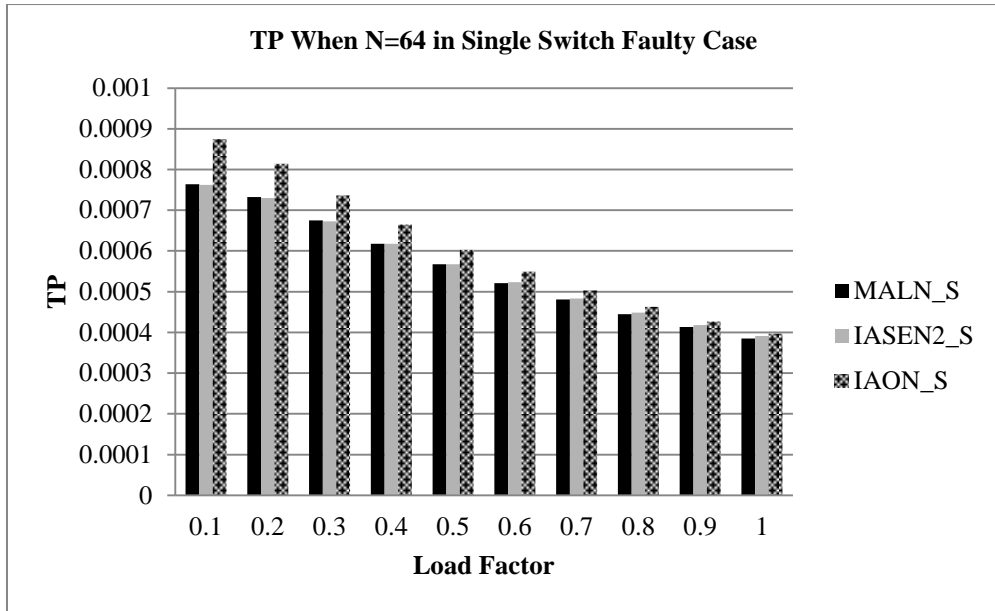


Figure 4.22. Comparison of TP for MALN, IASEN-2 and IAON when N=64 in Single Switch Faulty Condition.

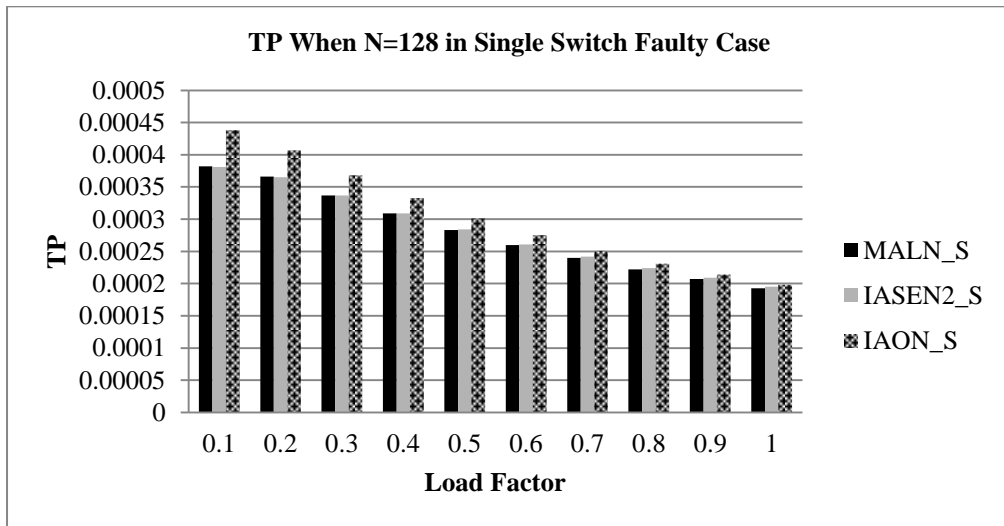


Figure 4.23. Comparison of TP for MALN, IASEN-2 and IAON when N=128 in Single Switch Faulty Condition.

4.5.12 Processor Utilization for MALN, IASEN-2 and IAON in Non-Faulty Condition

In figure 4.24, 4.25, 4.26 and 4.27, the author has shown the processor utilization of MALN, IASEN-2 and IAON for N=16, 32, 64 and 128 respectively. Here the processor utilization is calculated in non-faulty condition. In all figures, it is clear that the IAON has better processor utilization than the earlier proposed MINs.

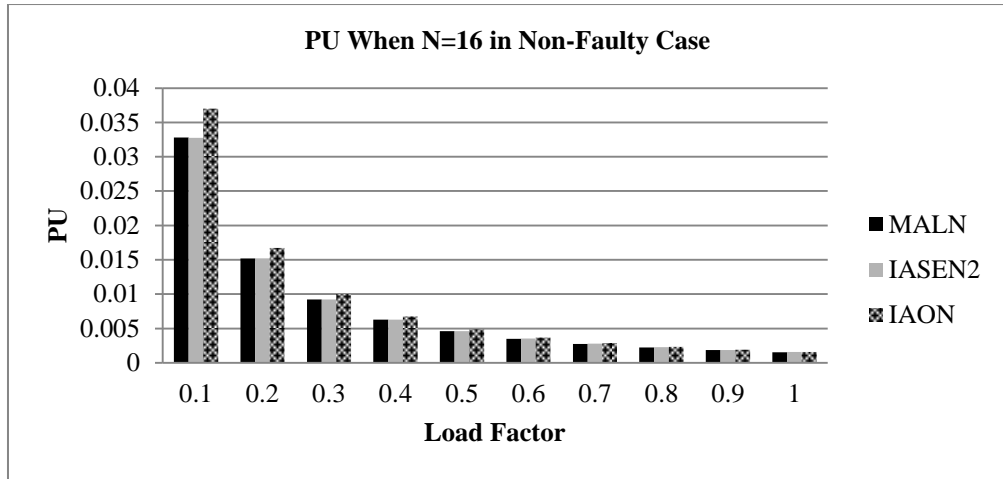


Figure 4.24. Comparison of PU for MALN, IASEN-2 and IAON when N=16 in Non-Faulty Condition.

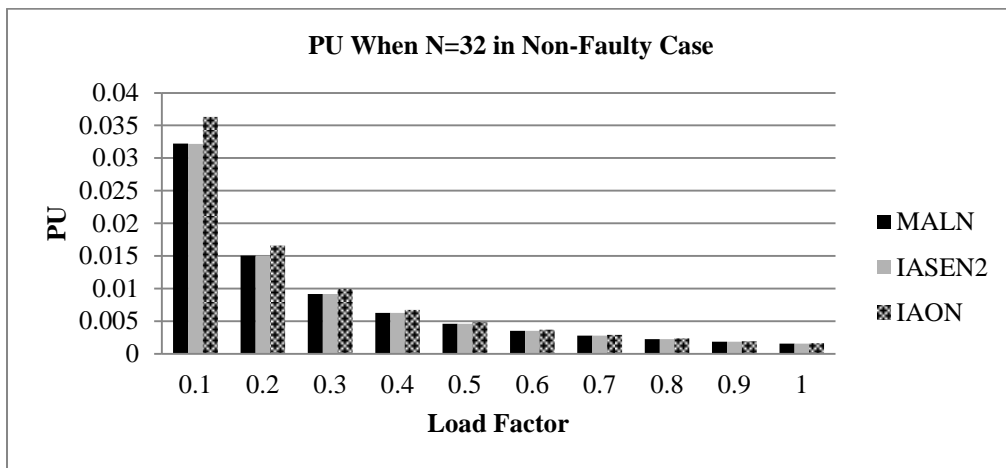


Figure 4.25. Comparison of PU for MALN, IASEN-2 and IAON when N=32 in Non-Faulty Condition.

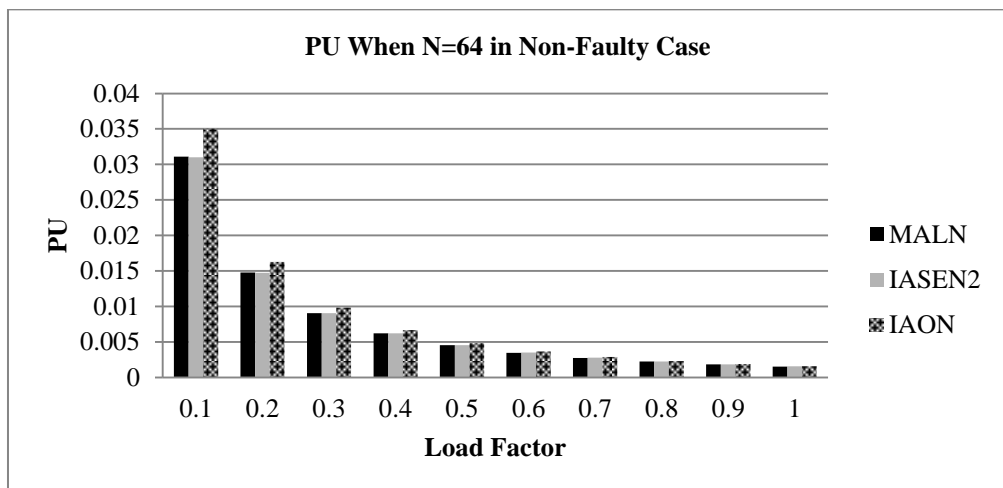


Figure 4.26. Comparison of PU for MALN, IASEN-2 and IAON when N=64 in Non-Faulty Condition.

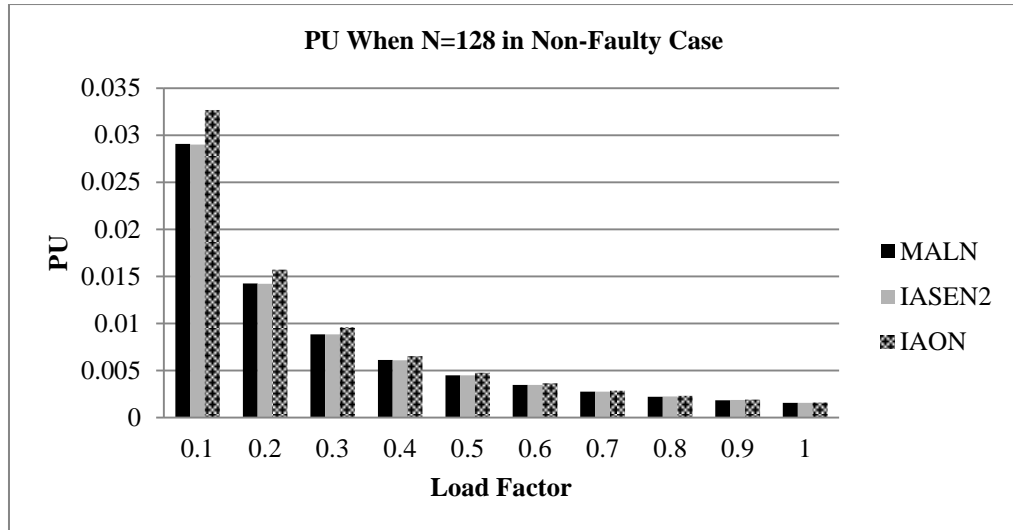


Figure 4.27. Comparison of PU for MALN, IASEN-2 and IAON when N=128 in Non-Faulty Condition.

4.5.13 Processor Utilization for MALN, IASEN-2 and IAON in Single Switch Faulty Condition

In figure 4.28, 4.29, 4.30 and 4.31, the author has shown the processor utilization of MALN, IASEN-2 and IAON for N=16, 32, 64 and 128 respectively. Here the processor utilization is calculated in single switch faulty condition. In all figures, it is clear that the IAON has better processor utilization than the earlier proposed MINs.

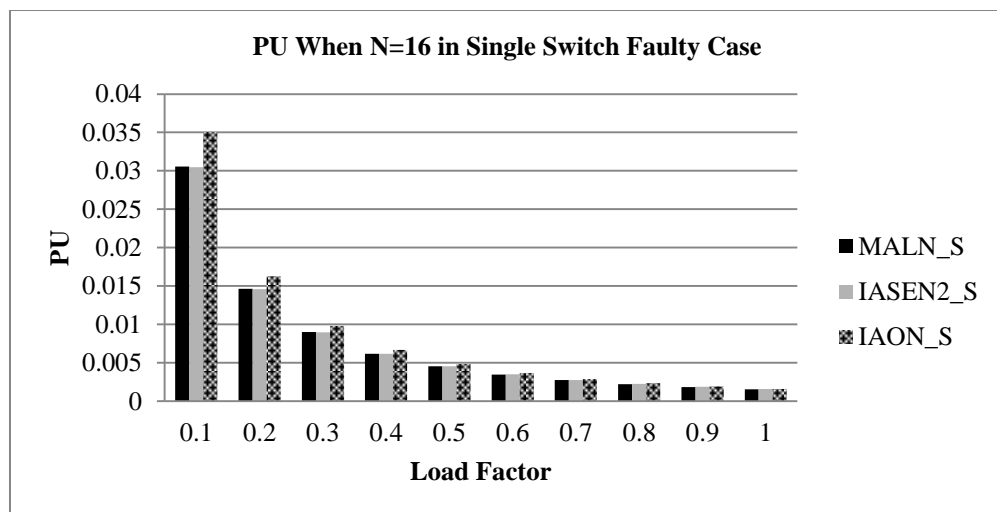


Figure 4.28. Comparison of PU for MALN, IASEN-2 and IAON when N=16 in Single Switch Faulty Condition.

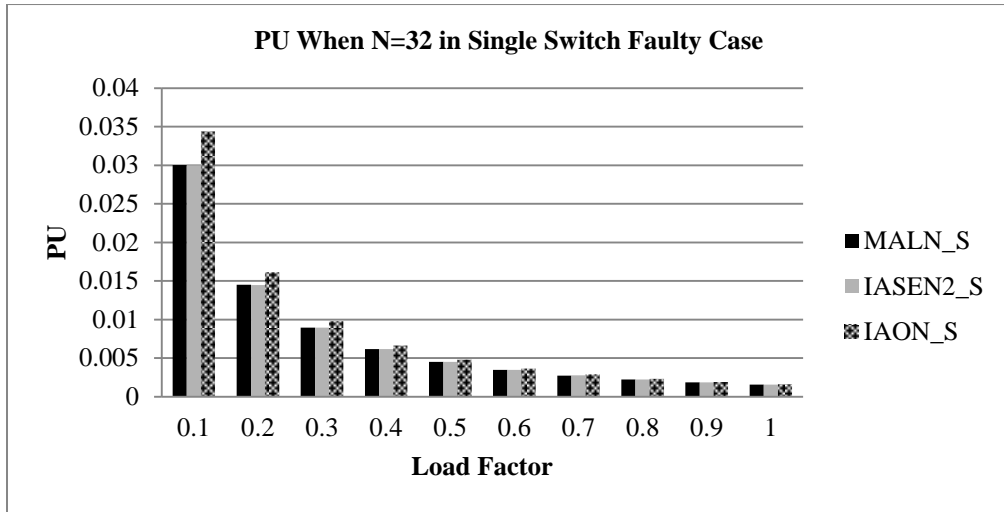


Figure 4.29. Comparison of PU for MALN, IASEN-2 and IAON when N=32 in Single Switch Faulty Condition.

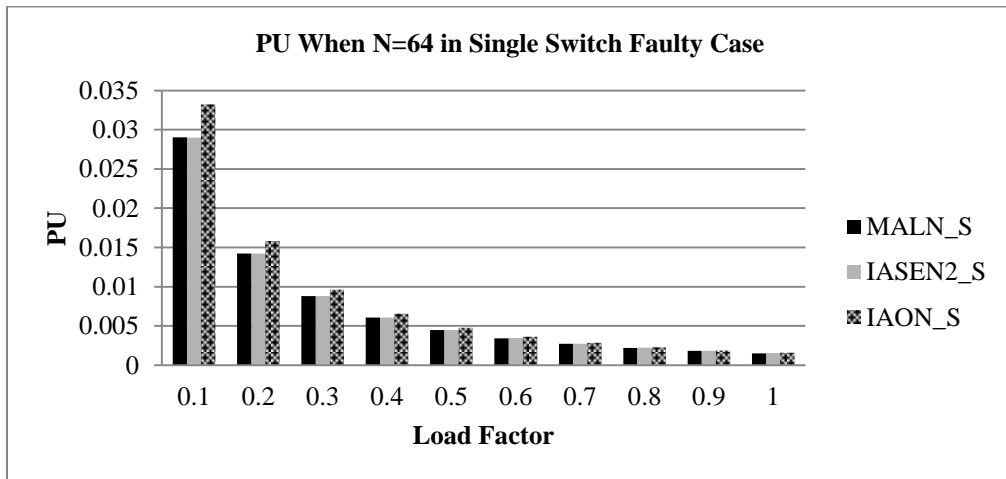


Figure 4.30. Comparison of PU for MALN, IASEN-2 and IAON when N=64 in Single Switch Faulty Condition.

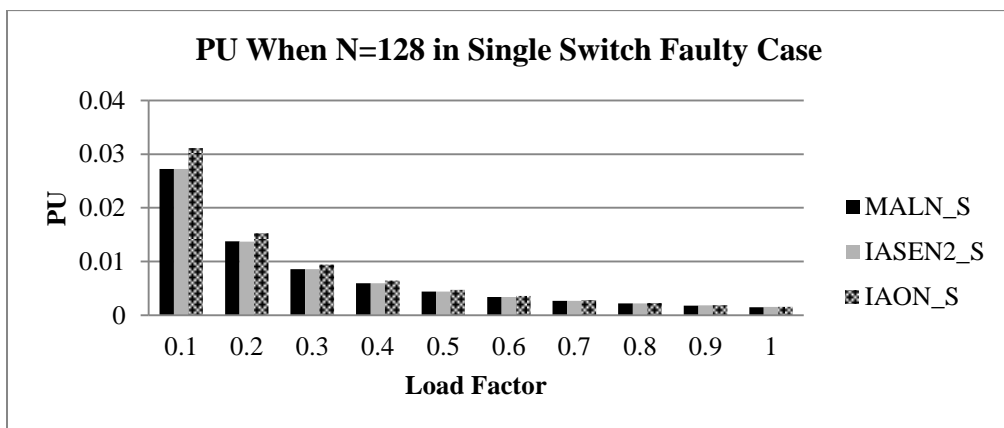


Figure 4.31. Comparison of PU for MALN, IASEN-2 and IAON when N=128 in Single Switch Faulty Condition.

4.5.14 Processing Power for MALN, IASEN-2 and IAON in Non-Faulty Condition

In figure 4.32, 4.33, 4.34 and 4.35, the author has shown the processing power of MALN, IASEN-2 and IAON for N=16, 32, 64 and 128 respectively. Here the processor utilization is calculated in non-faulty condition. In all figures, it is clear that the IAON has better processing power than the earlier proposed MINs.

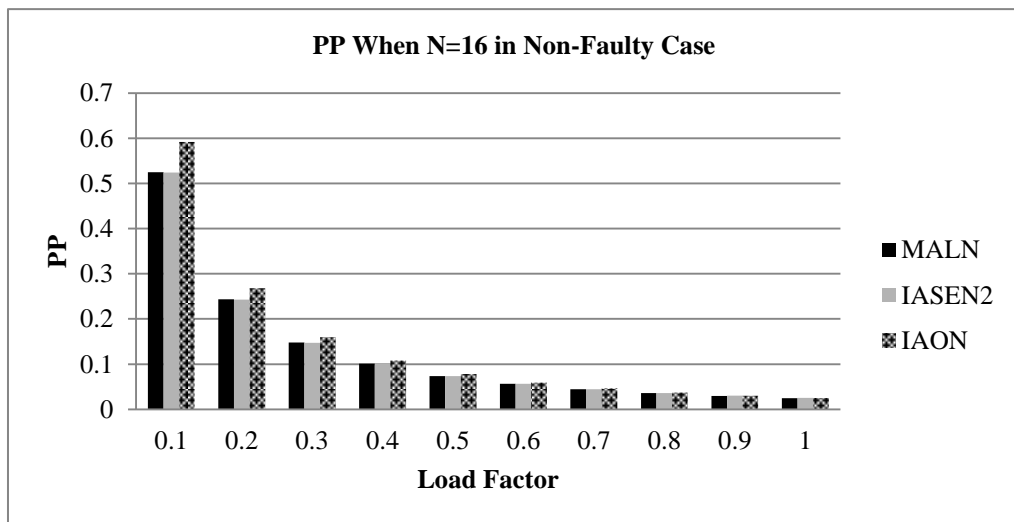


Figure 4.32. Comparison of PP for MALN, IASEN-2 and IAON when N=16 in Non-Faulty Condition.

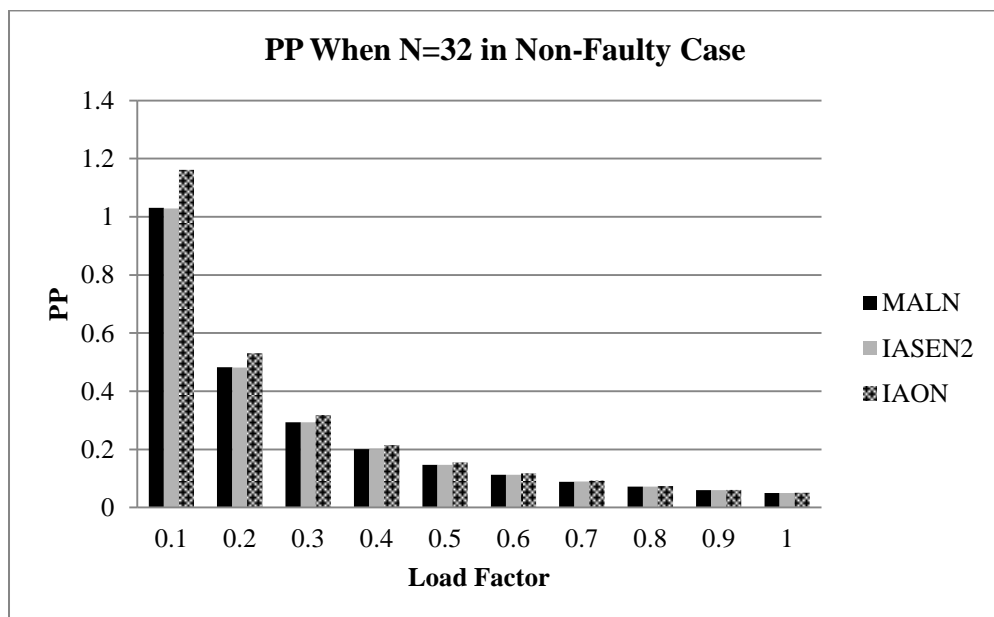


Figure 4.33. Comparison of PP for MALN, IASEN-2 and IAON when N=32 in Non-Faulty Condition.

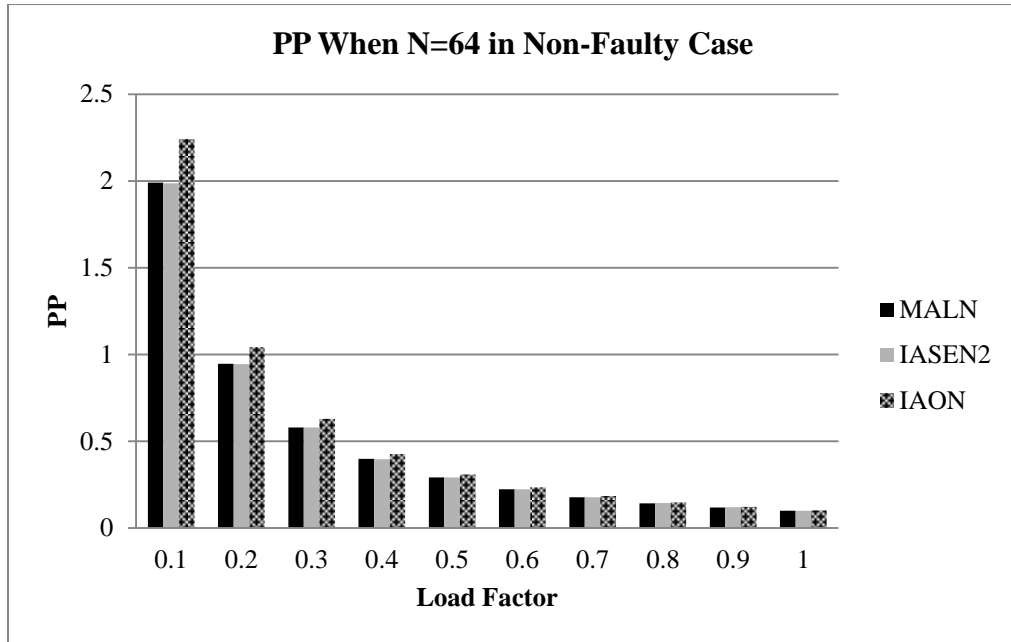


Figure 4.34. Comparison of PP for MALN, IASEN-2 and IAON when N=64 in Non-Faulty Condition.

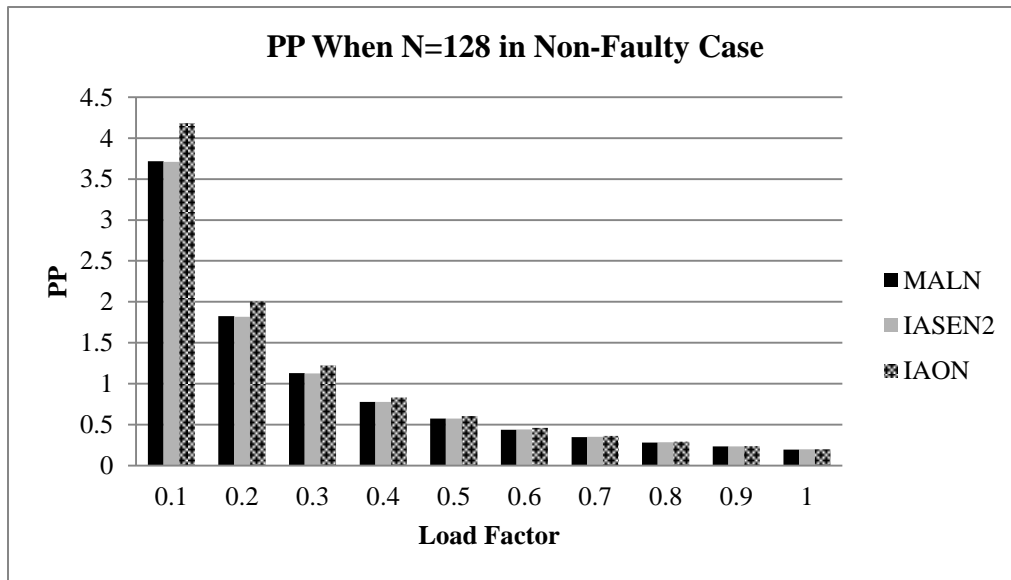


Figure 4.35. Comparison of PP for MALN, IASEN-2 and IAON when N=128 in Non-Faulty Condition.

4.5.15 Processing Power for MALN, IASEN-2 and IAON in Single Switch Faulty Condition

In figure 4.36, 4.37, 4.38 and 4.39, the author has shown the processing power of MALN, IASEN-2 and IAON for N=16, 32, 64 and 128 respectively. Here the processor utilization is calculated in single switch faulty condition.

In all figures, it is clear that the IAON has better processing power than the earlier proposed MINs.

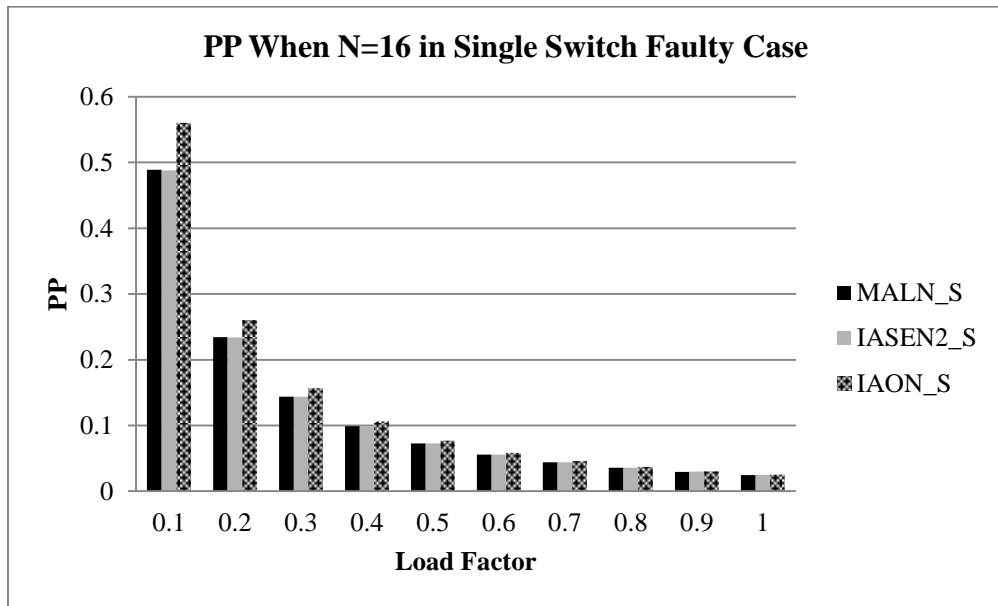


Figure 4.36. Comparison of PP for MALN, IASEN-2 and IAON when N=16 in Single Switch Faulty Condition.

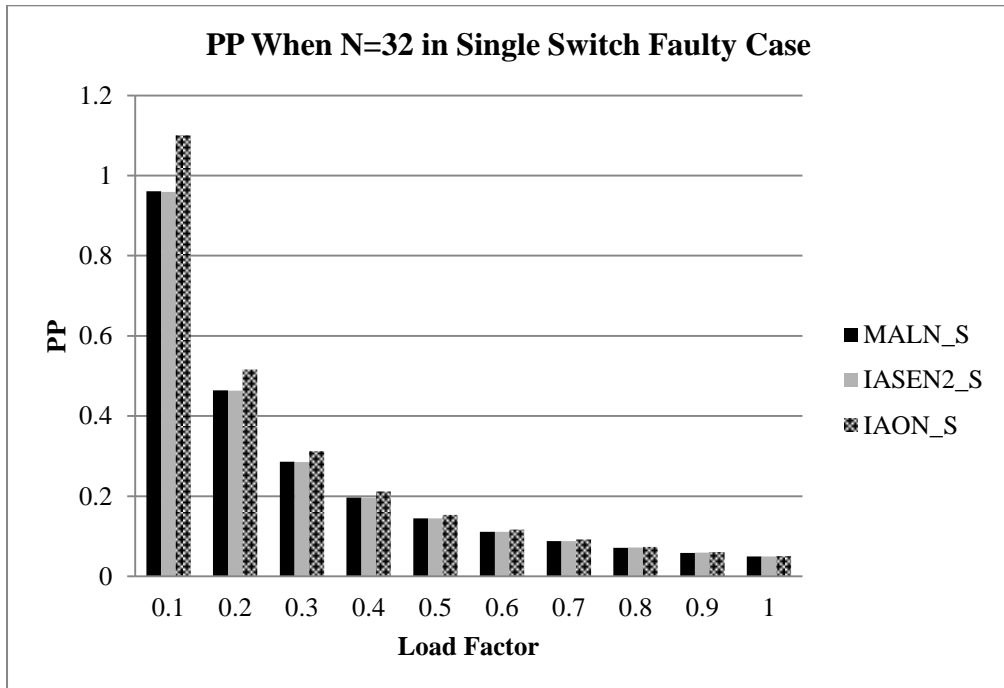


Figure 4.37. Comparison of PP for MALN, IASEN-2 and IAON when N=32 in Single Switch Faulty Condition.

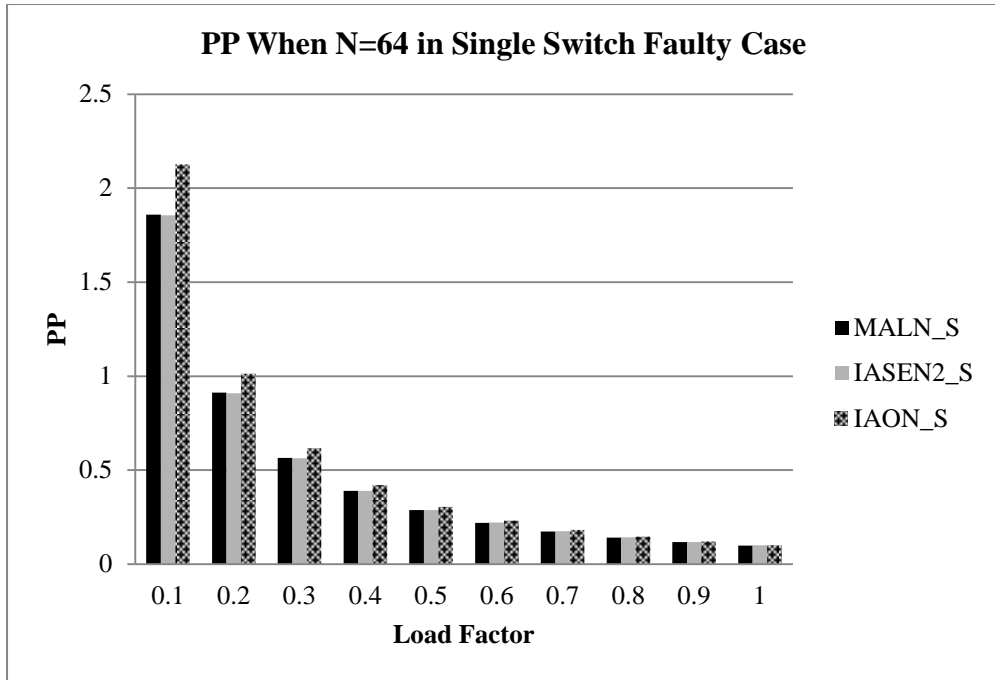


Figure 4.38. Comparison of PP for MALN, IASEN-2 and IAON when N=64 in Single Switch Faulty Condition.

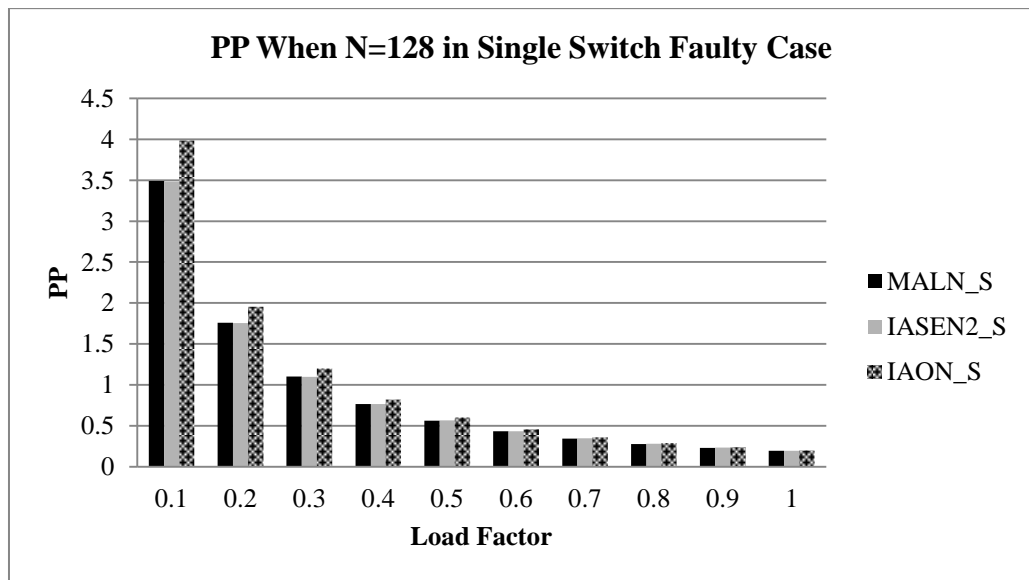


Figure 4.39. Comparison of PP for MALN, IASEN-2 and IAON when N=128 in Single Switch Faulty Condition.

Here the author has shown the simulation results of MALN, IASEN-2 and IAON (from figure 4.11 to 4.39). Further he calculated the average PA which is as follows:

Network	PA
MALN	0.561
IASEN-2	0.564
IAON	0.588

Table 4.4

The table 4.4 clearly shows that the PA of MALN and IASEN-2 are 56.1% and 56.4% whereas PA of IAON is 58.8%. It means IAON is 2.7% better than MALN and 2.4% better than IASEN-2 in terms of PA.

Further, it is observed that performance of a network is primarily depending on the bandwidth because most of the factors of performance (e.g. TP, PU, PP) are calculated on the basis of bandwidth. Therefore, the author has calculated the average bandwidth which is follows:

N	MALN	IASEN-2	IAON
16	8.983	9.032	9.420
32	17.966	18.065	18.840
64	35.933	36.130	37.680
128	71.866	72.261	75.360

Table 4.5

From table 4.5, it is observed that when size of network is doubled then value of bandwidth also gets doubled. So,

For MALN=>

$$\left(\frac{8.983}{16}\right) \times 100 = 56.1\%$$

Here the size of network is 16 hence the author has taken the value 16 in denominator and multiplied by 100 to calculate the percentage.

For IASEN-2=>

$$\left(\frac{9.032}{16}\right) \times 100 = 56.4\%$$

For IAON=>

$$\left(\frac{9.420}{16}\right) \times 100 = 58.8\%$$

Again, it is observed that IAON is 2.7% better than MALN and 2.4% better than IASEN-2.

4.5.16 Bandwidth for IASEN-2, IAON and IASEN-3

In figure 4.40, 4.41, 4.42 and 4.43, the author has shown the bandwidth of IASEN-2, IAON and IASEN-3 for N=16, 32, 64 and 128 respectively. In all figures, it is clear that the IASEN-3 has better bandwidth than the IASEN-2 and IAON.

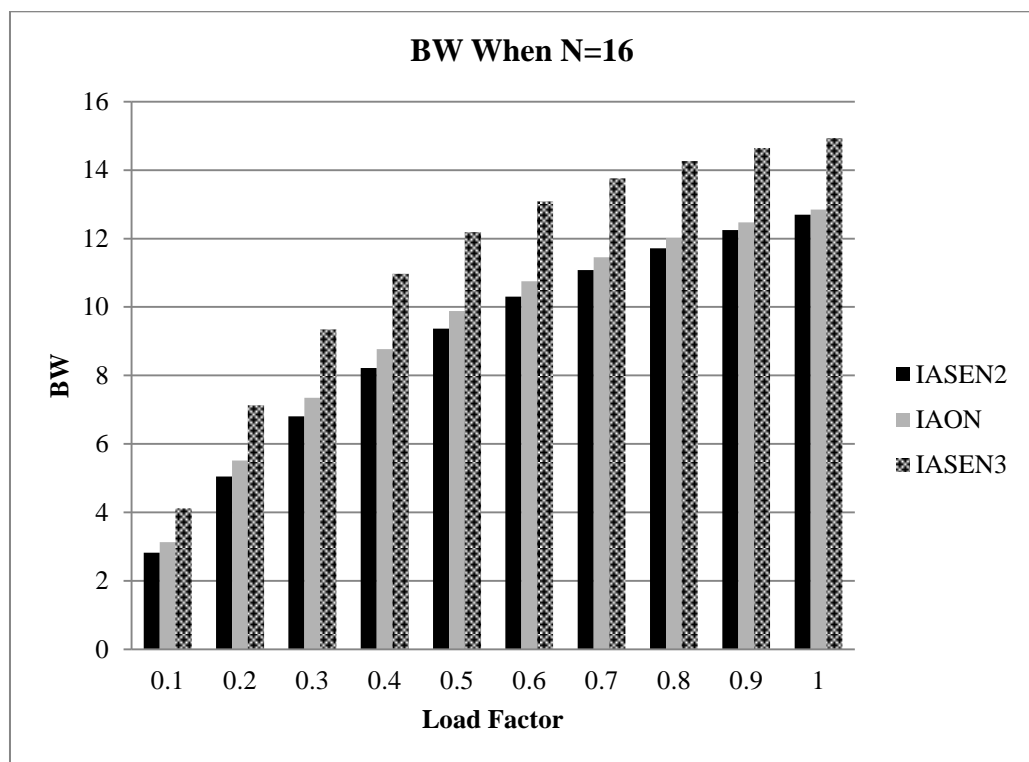


Figure 4.40. Comparison of BW for IASEN-2, IAON and IASEN-3 when N=16.

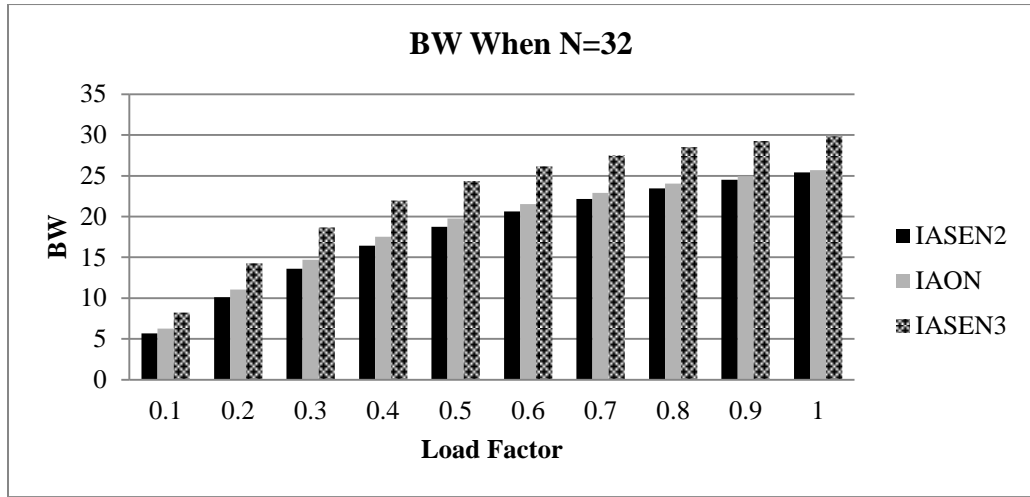


Figure 4.41. Comparison of BW for IASEN-2, IAON and IASEN-3 when N=32.

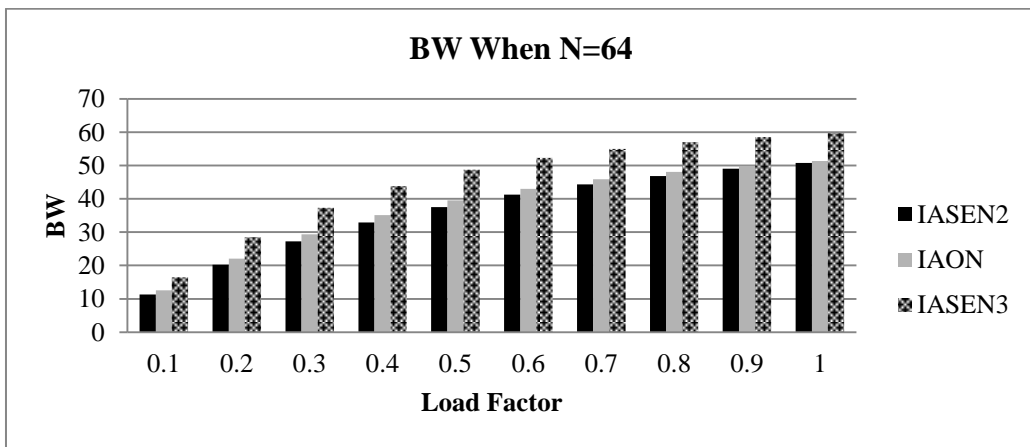


Figure 4.42. Comparison of BW for IASEN-2, IAON and IASEN-3 when N=64.

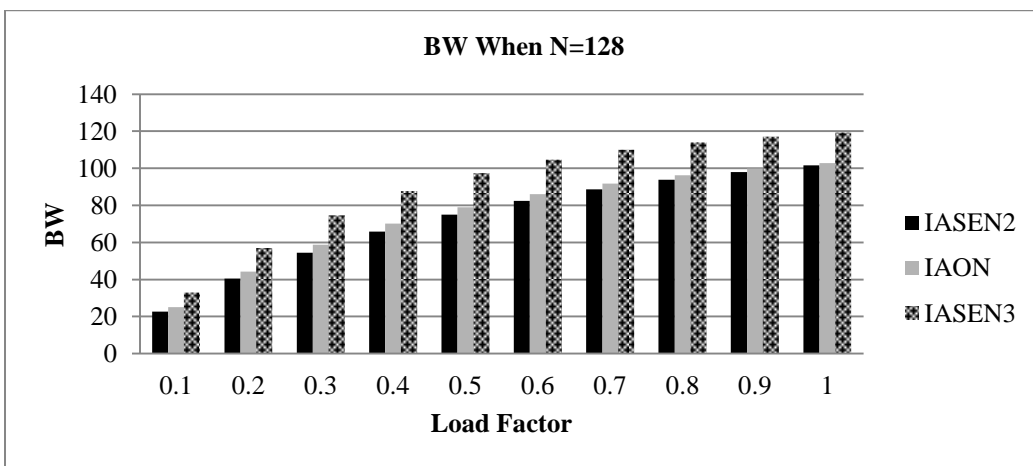


Figure 4.43. Comparison of BW for IASEN-2, IAON and IASEN-3 when N=128.

4.5.17 Probability of Acceptance for IASEN-2, IAON and IASEN-3

In figure 4.44, the author has shown the bandwidth of IASEN-2, IAON and IASEN-3 for N=16, 32, 64 and 128 respectively. In all figures, it is clear that the IASEN-3 has a better probability of acceptance than the IASEN-2 and IAON.

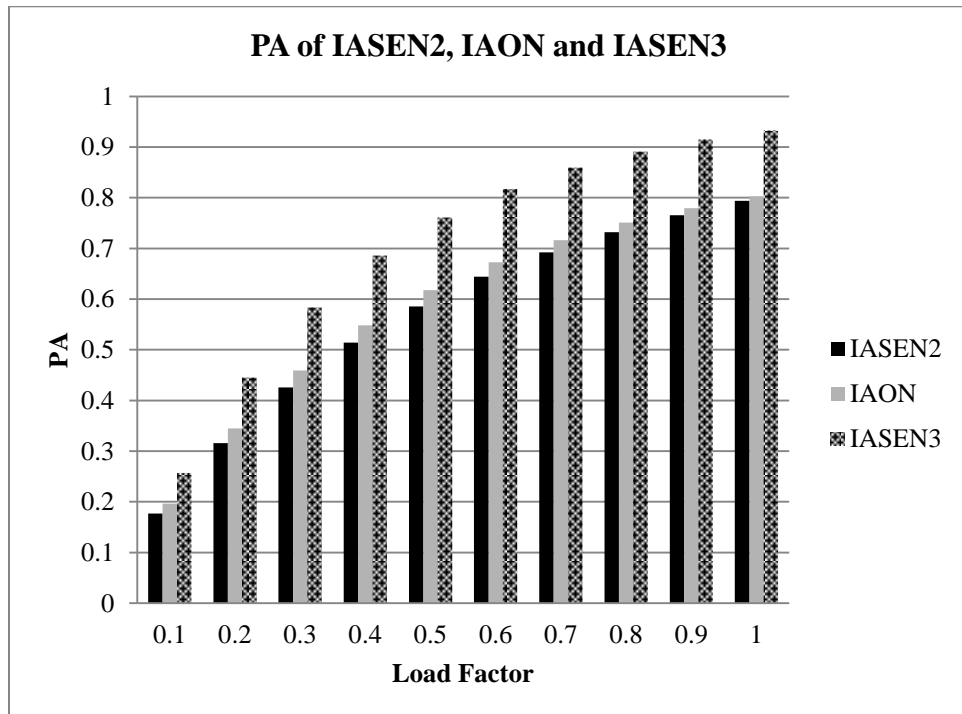


Figure 4.44. Comparison of PA for IASEN-2, IAON and IASEN-3.

4.5.18 Throughput for IASEN-2, IAON and IASEN-3 in Non-Faulty Condition

In figure 4.45, 4.46, 4.47 and 4.48, the author has shown the throughput of IASEN-2, IAON and IASEN-3 for N=16, 32, 64 and 128 respectively. Here the throughput is calculated in non-faulty condition. In all figures, it is clear that the IASEN-3 has better throughput than the IASEN-2 and IAON.

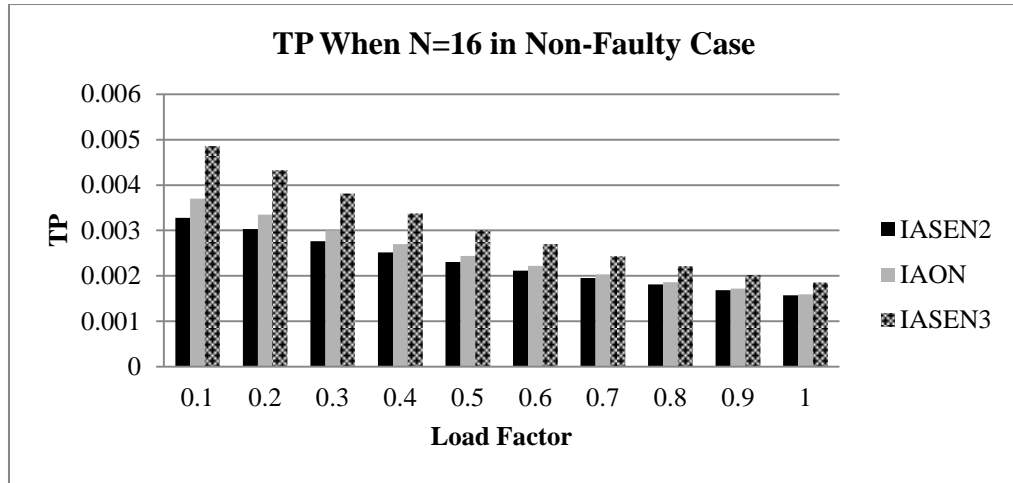


Figure 4.45. Comparison of TP for IASSEN-2, IAON and IASSEN-3 when N=16 in Non-Faulty Condition.

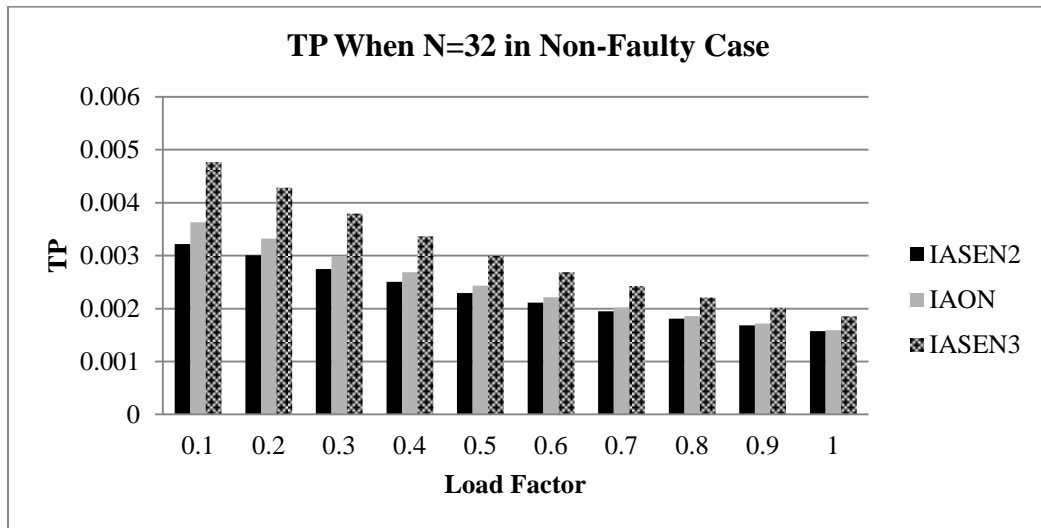


Figure 4.46. Comparison of TP for IASSEN-2, IAON and IASSEN-3 when N=32 in Non-Faulty Condition.

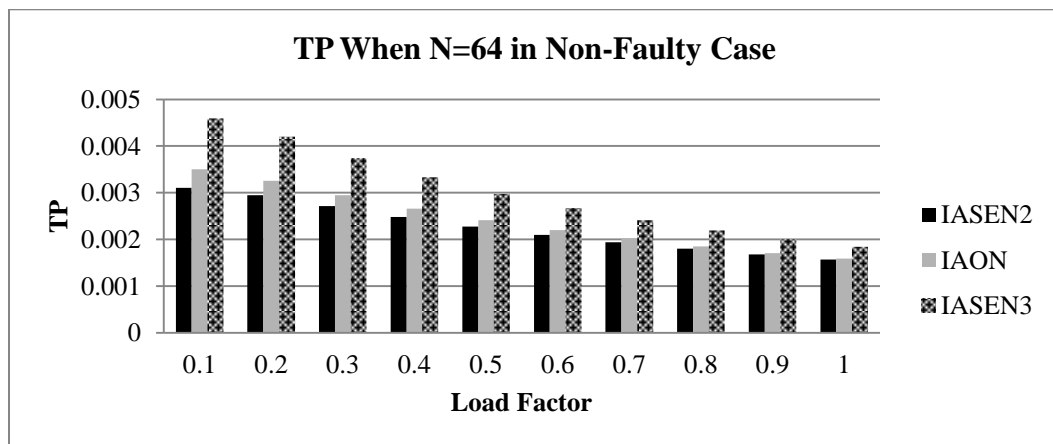


Figure 4.47. Comparison of TP for IASSEN-2, IAON and IASSEN-3 when N=64 in Non-Faulty Condition.

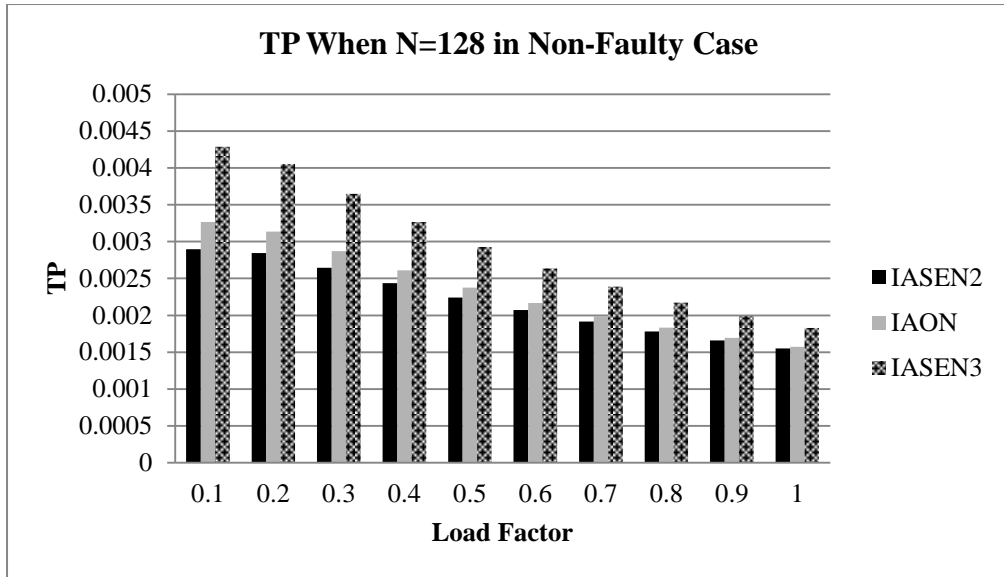


Figure 4.48. Comparison of TP for IASEN-2, IAON and IASEN-3 when N=128 in Non-Faulty Condition.

4.5.19 Throughput for IASEN-2, IAON and IASEN-3 in Single Switch Faulty Condition

In figure 4.49, 4.50, 4.51 and 4.52, the author has shown the throughput of IASEN-2, IAON and IASEN-3 for N=16, 32, 64 and 128 respectively. Here the throughput is calculated in single switch faulty condition. In all figures, it is clear that the IASEN-3 has better throughput than the IASEN-2 and IAON.

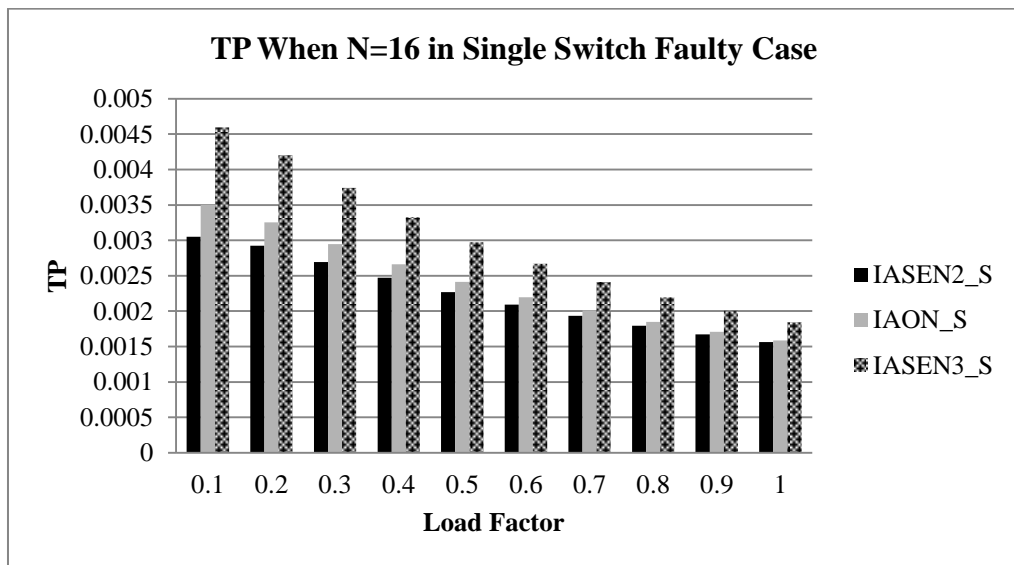


Figure 4.49. Comparison of TP for IASEN-2, IAON and IASEN-3 when N=16 in Single Switch Faulty Condition.

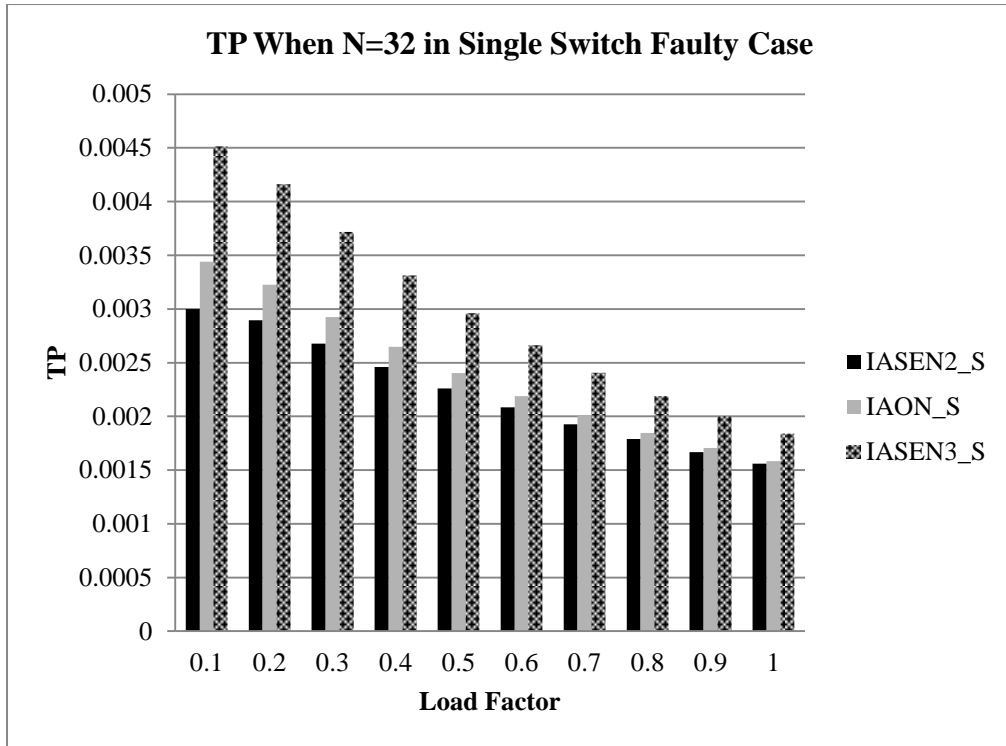


Figure 4.50. Comparison of TP for IASSEN-2, IAON and IASSEN-3 when N=32 in Single Switch Faulty Condition.

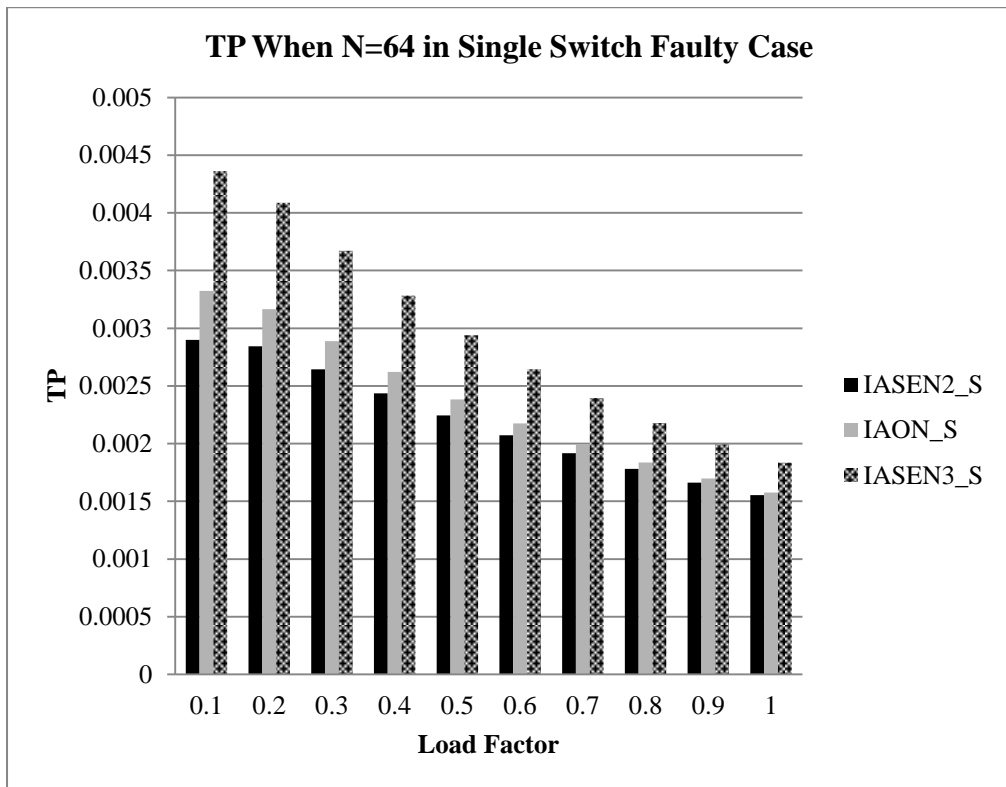


Figure 4.51. Comparison of TP for IASSEN-2, IAON and IASSEN-3 when N=64 in Single Switch Faulty Condition.

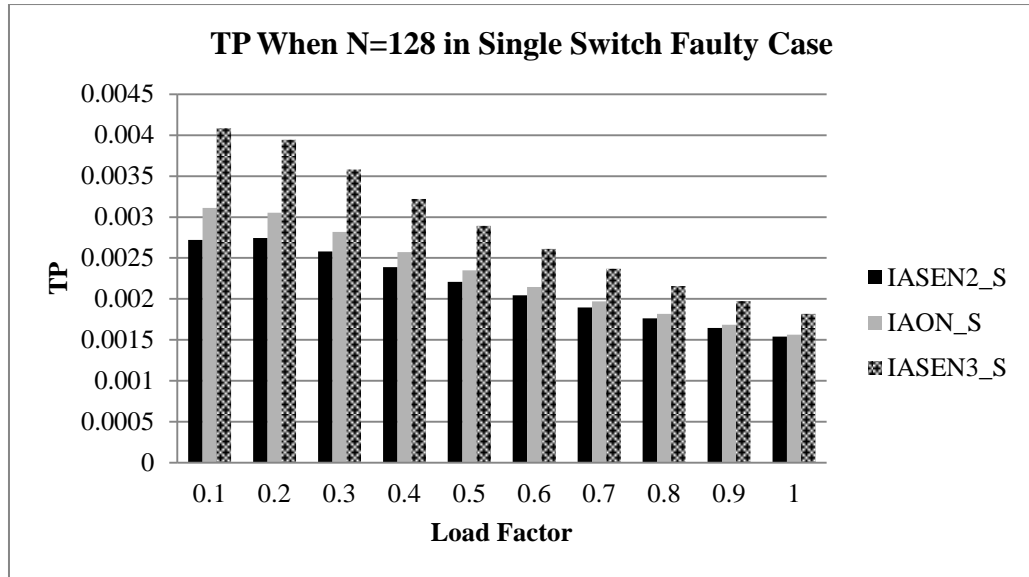


Figure 4.52. Comparison of TP for IASEN-2, IAON and IASEN-3 when N=128 in Single Switch Faulty Condition.

4.5.20 Processor Utilization for IASEN-2, IAON and IASEN-3 in Non-Faulty Condition

In figure 4.53, 4.54, 4.55 and 4.56, the author has shown the processor utilization of IASEN-2, IAON and IASEN-3 for N=16, 32, 64 and 128 respectively. Here the processor utilization is calculated in non-faulty condition. In all figures, it is clear that the IASEN-3 has better processor utilization than the IASEN-2 and IAON.

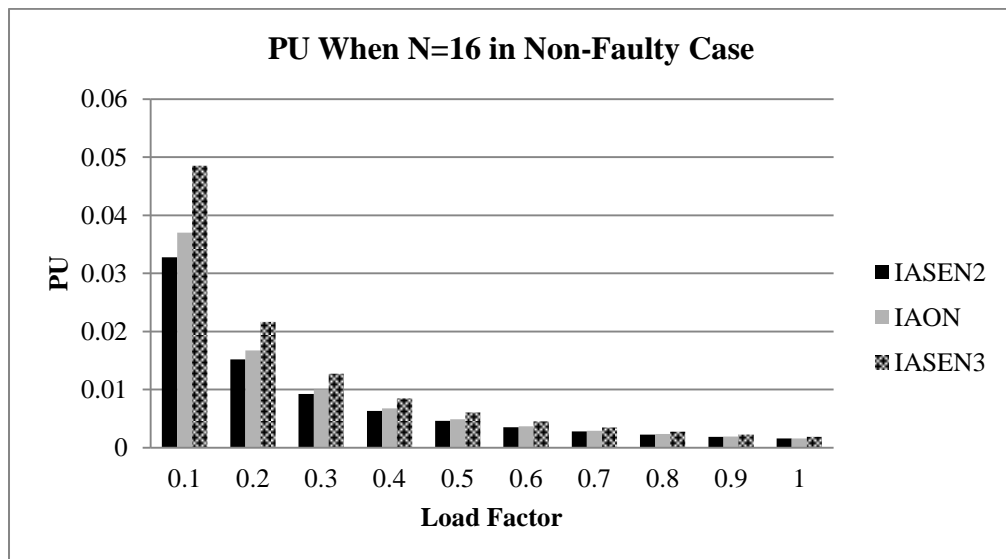


Figure 4.53. Comparison of PU for IASEN-2, IAON and IASEN-3 when N=16 in Non-Faulty Condition.

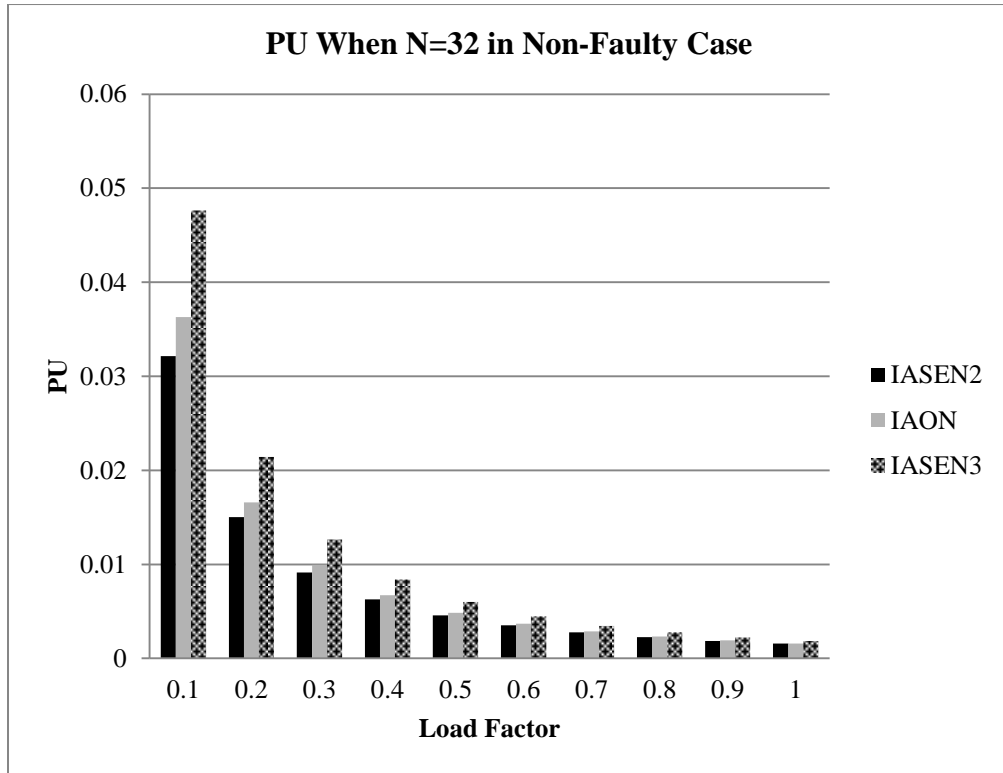


Figure 4.54. Comparison of PU for IASSEN-2, IAON and IASSEN-3 when N=32 in Non-Faulty Condition.

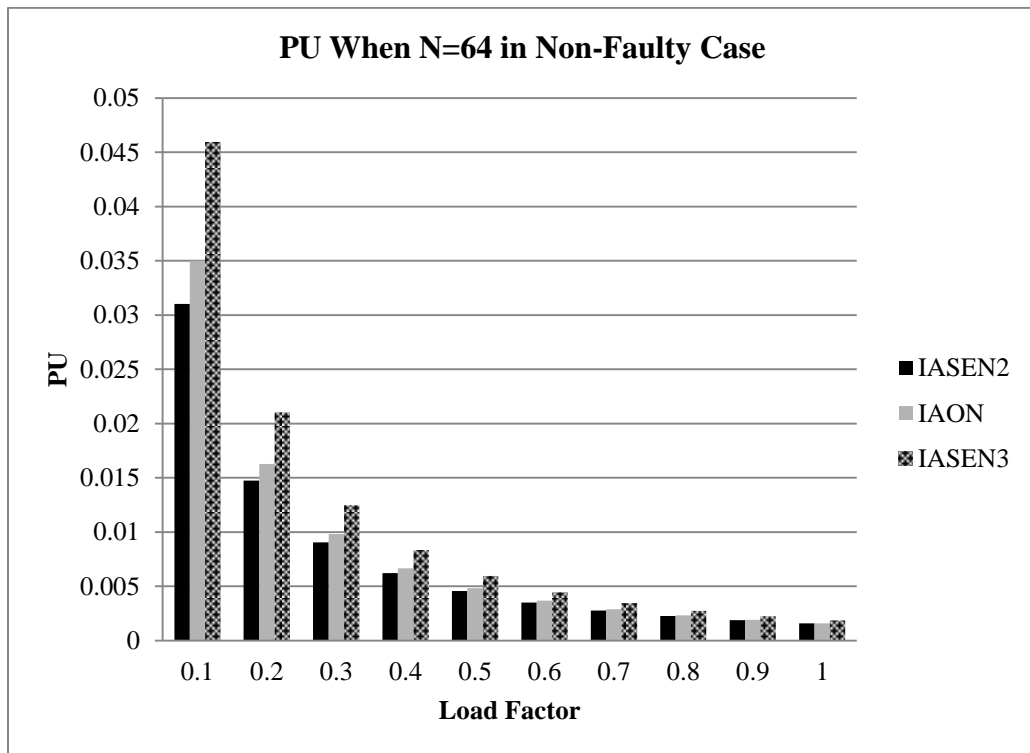


Figure 4.55. Comparison of PU for IASSEN-2, IAON and IASSEN-3 when N=64 in Non-Faulty Condition.

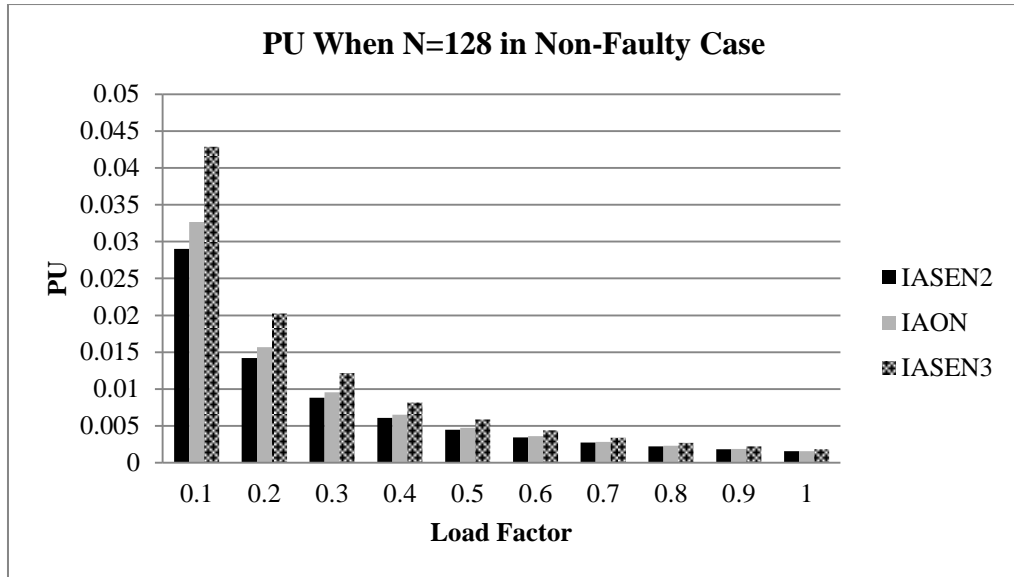


Figure 4.56. Comparison of PU for IASSEN-2, IAON and IASSEN-3 when N=128 in Non-Faulty Condition.

4.5.21 Processor Utilization for IASSEN-2, IAON and IASSEN-3 in Single Switch Faulty Condition

In figure 4.57, 4.58, 4.59 and 4.60, the author has shown the processor utilization of IASSEN-2, IAON and IASSEN-3 for N=16, 32, 64 and 128 respectively. Here the processor utilization is calculated in single switch faulty condition. In all figures, it is clear that the IASSEN-3 has better processor utilization than the IASSEN-2 and IAON.

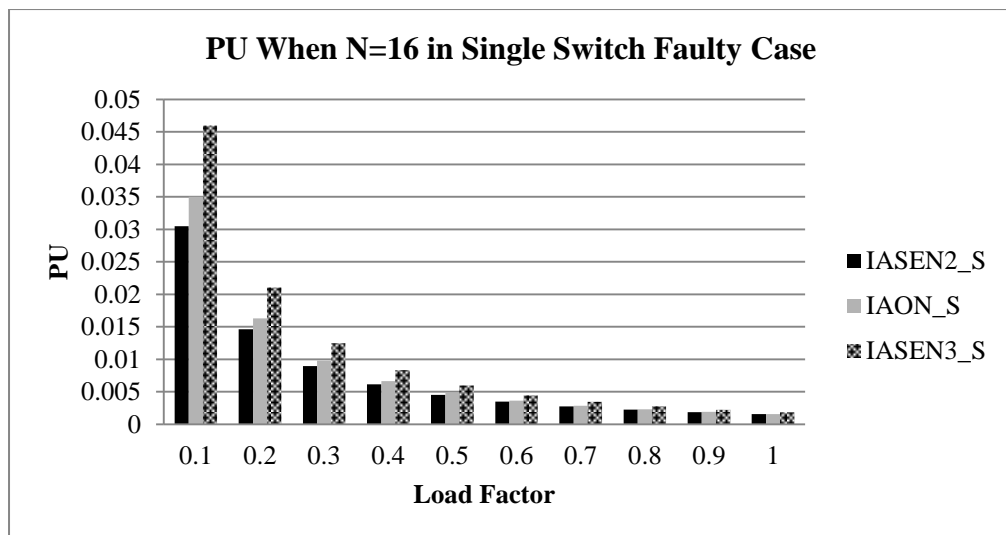


Figure 4.57. Comparison of PU for IASSEN-2, IAON and IASSEN-3 when N=16 in Single Switch Faulty Condition.

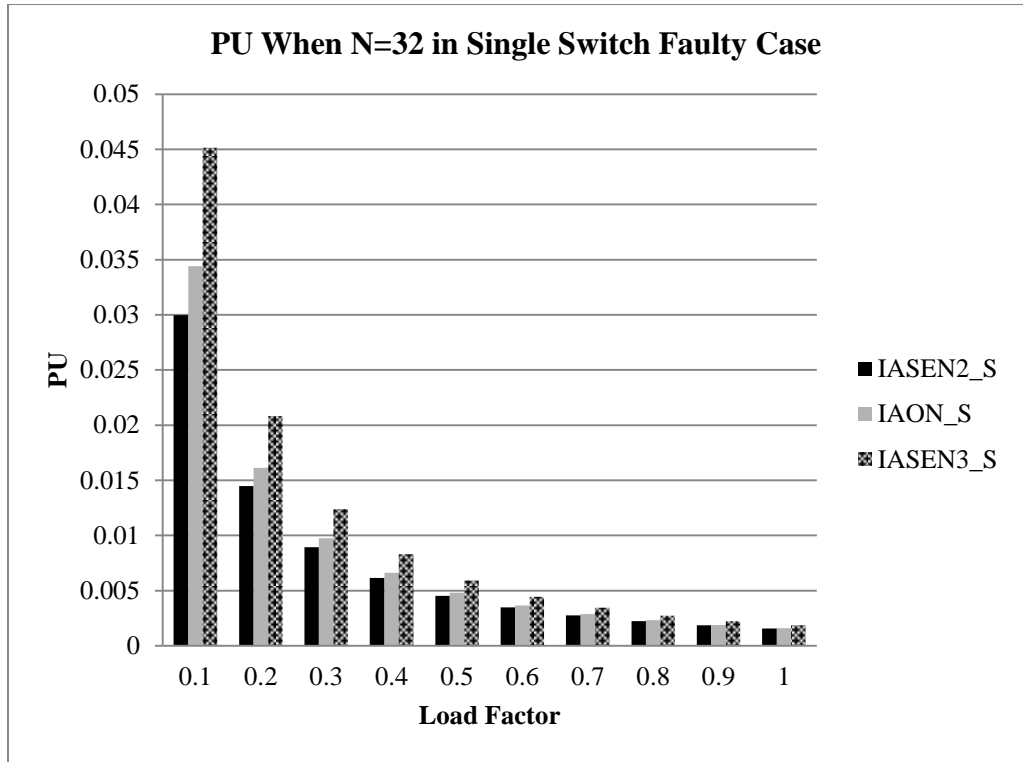


Figure 4.58. Comparison of PU for IASEN-2, IAON and IASEN-3 when N=32 in Single Switch Faulty Condition.

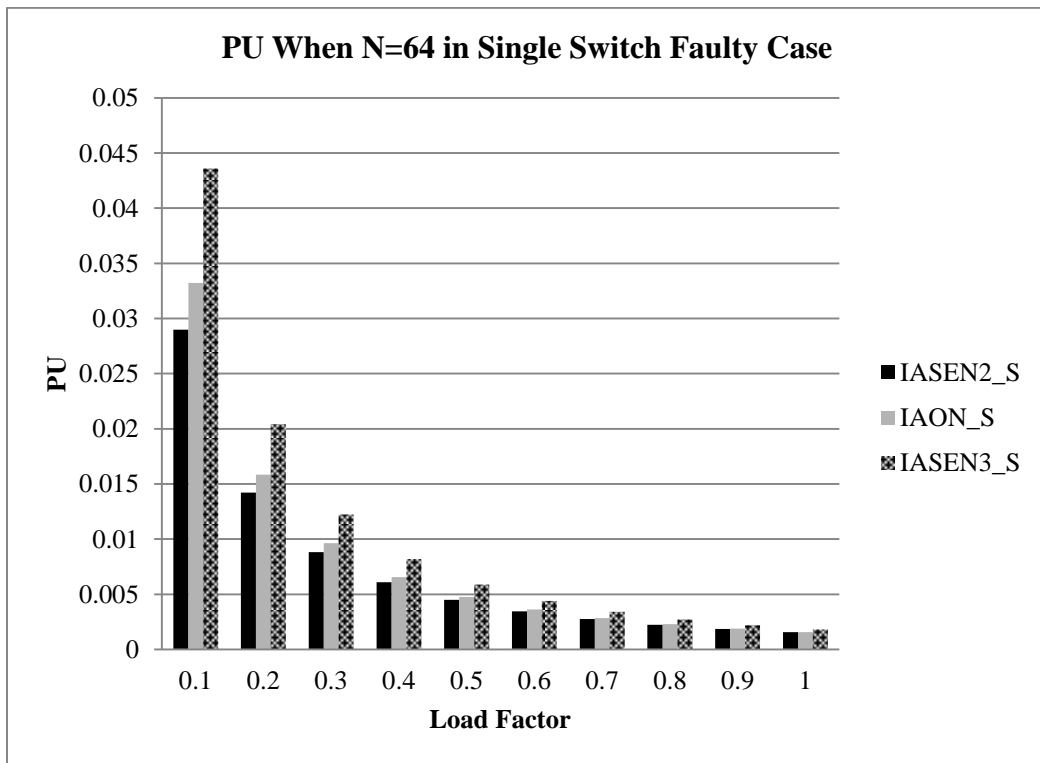


Figure 4.59. Comparison of PU for IASEN-2, IAON and IASEN-3 when N=64 in Single Switch Faulty Condition.

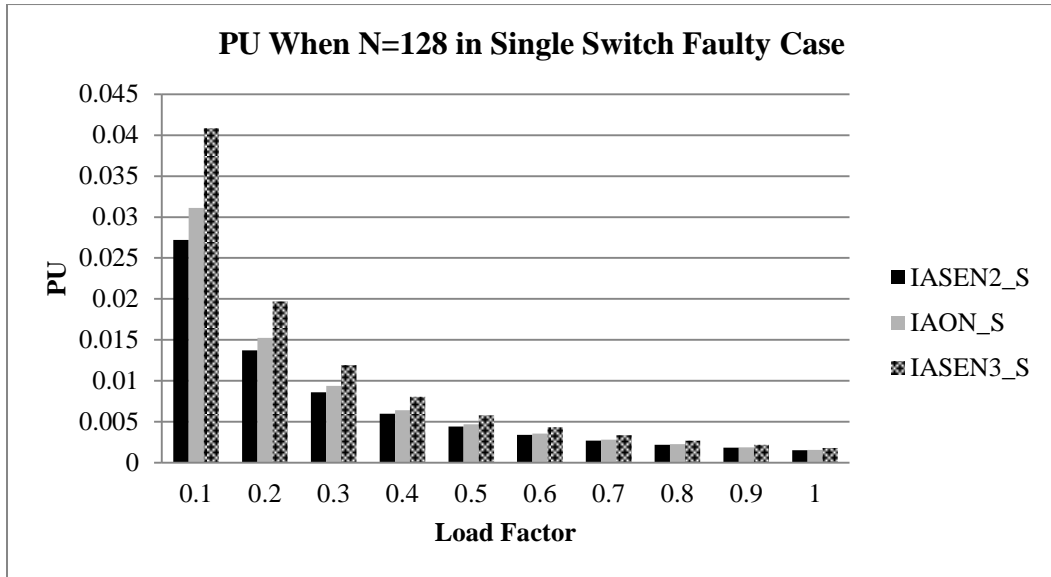


Figure 4.60. Comparison of PU for IASSEN-2, IAON and IASSEN-3 when N=128 in Single Switch Faulty Condition.

4.5.22 Processing Power for IASSEN-2, IAON and IASSEN-3 in Non-Faulty Condition

In figure 4.61, 4.62, 4.63 and 4.64, the author has shown the processor utilization of IASSEN-2, IAON and IASSEN-3 for N=16, 32, 64 and 128 respectively. Here the processor utilization is calculated in non-faulty condition. In all figures, it is clear that the IASSEN-3 has better processor utilization than the IASSEN-2 and IAON.

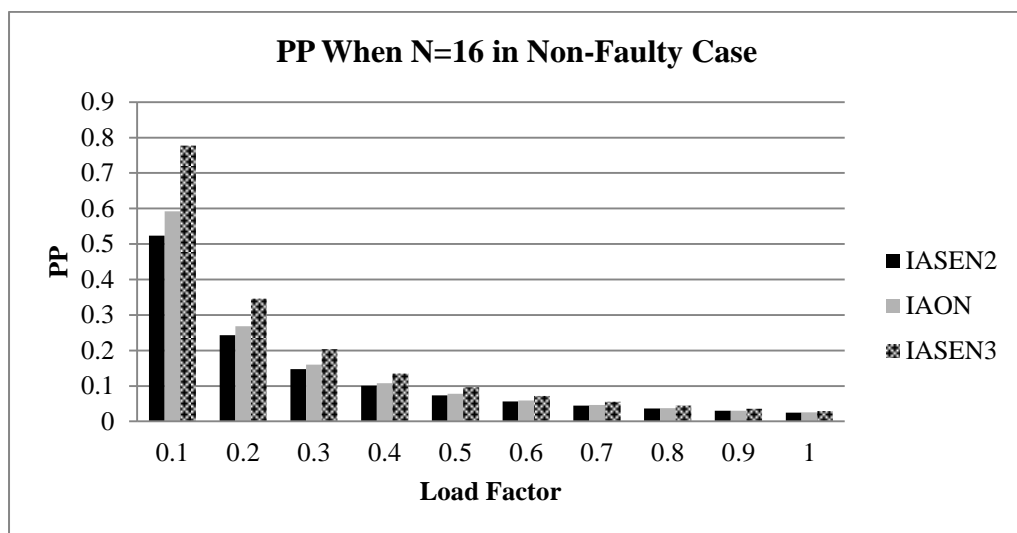


Figure 4.61. Comparison of PP for IASSEN-2, IAON and IASSEN-3 when N=16 in Non-Faulty Condition.

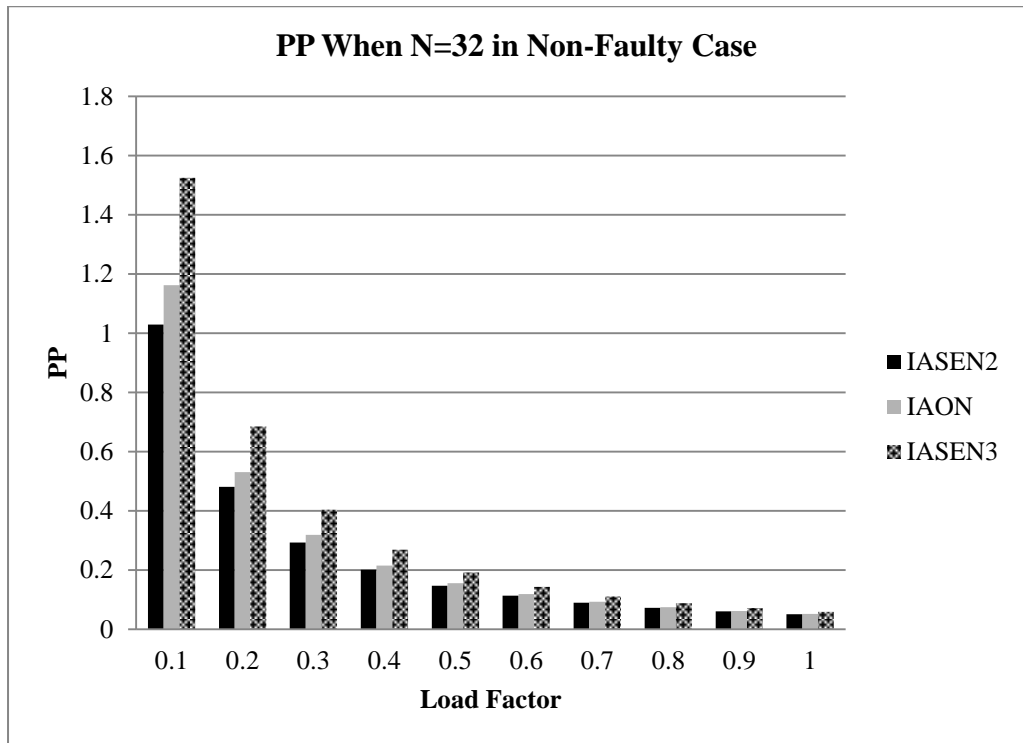


Figure 4.62. Comparison of PP for IASSEN-2, IAON and IASSEN-3 when N=32 in Non-Faulty Condition.

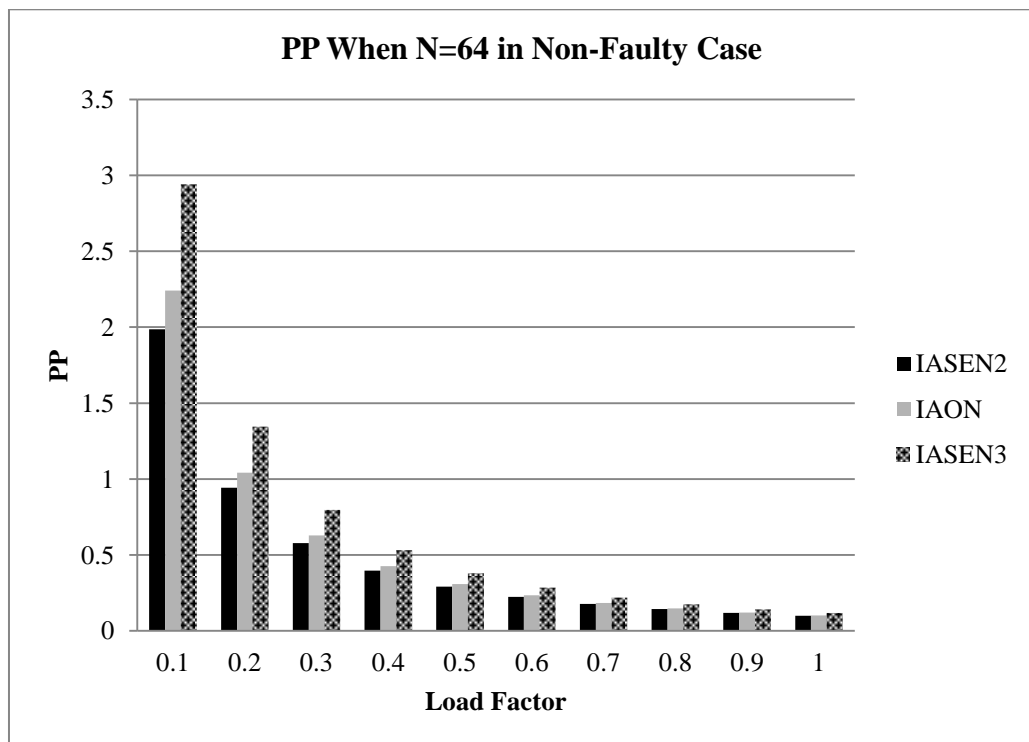


Figure 4.63. Comparison of PP for IASSEN-2, IAON, and IASSEN-3 when N=64 in Non-Faulty Condition.

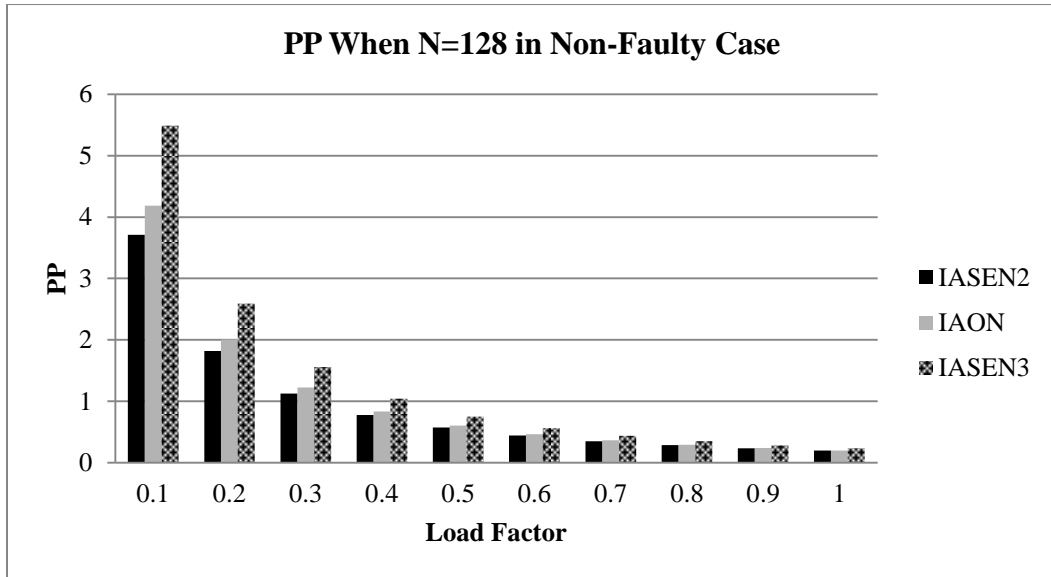


Figure 4.64. Comparison of PP for IASEN-2, IAON and IASEN-3 when N=128 in Non-Faulty Condition.

4.5.23 Processing Power for IASEN-2, IAON and IASEN-3 in Single Switch Faulty Condition

In figure 4.65, 4.66, 4.67 and 4.68, the author has shown the processor utilization of IASEN-2, IAON and IASEN-3 for N=16, 32, 64 and 128 respectively. Here the processor utilization is calculated in single switch faulty condition. In all figures, it is clear that the IASEN-3 has better processor utilization than the IASEN-2 and IAON.

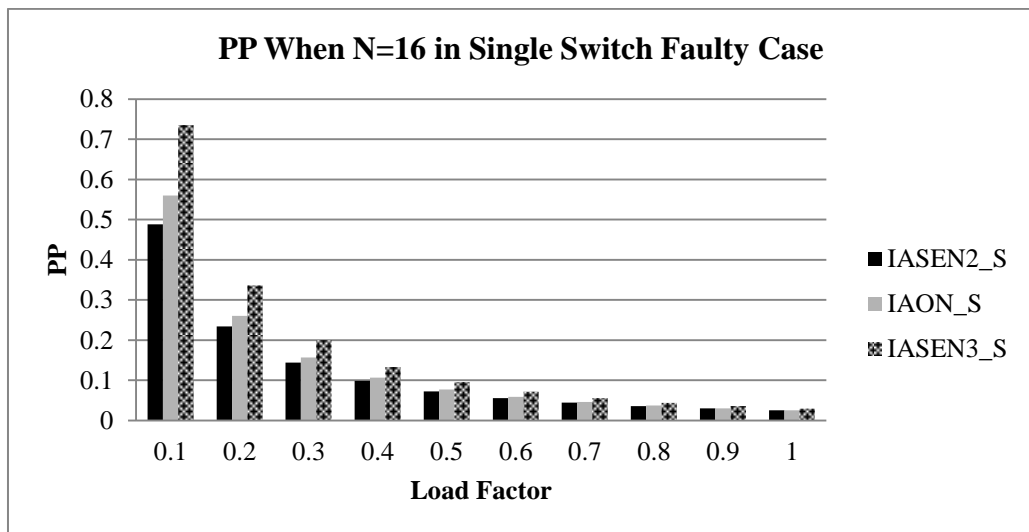


Figure 4.65. Comparison of PP for IASEN-2, IAON and IASEN-3 when N=16 in Single Switch Faulty Condition.

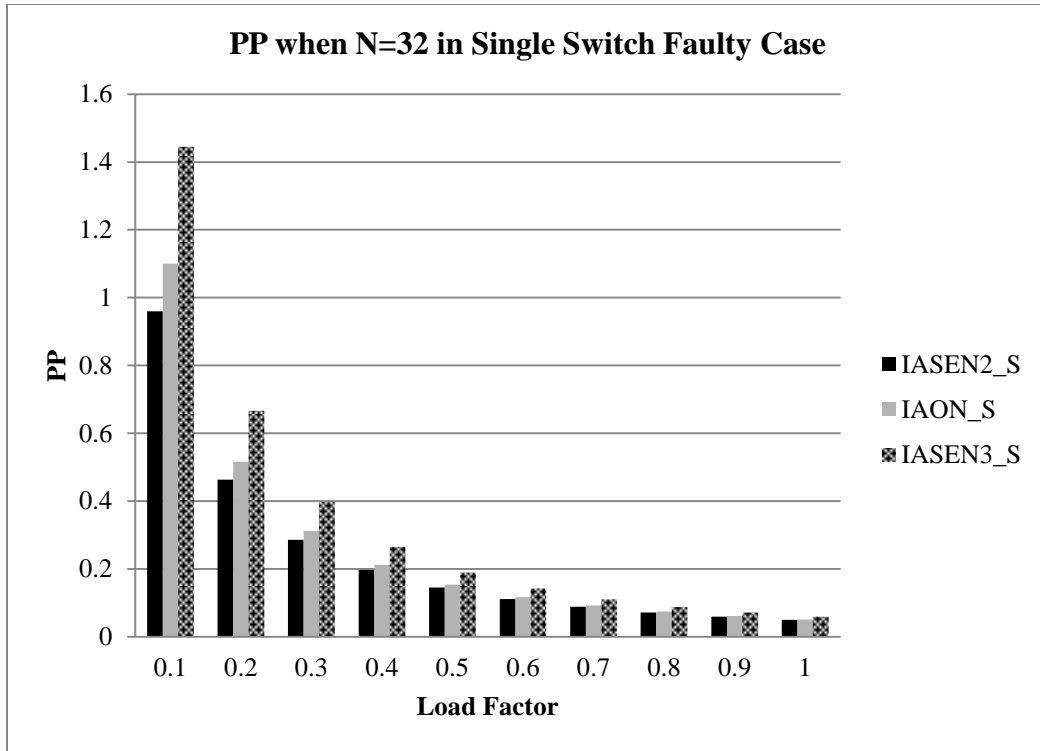


Figure 4.66. Comparison of PP for IASen-2, IAON and IASen-3 when N=32 in Single Switch Faulty Condition.

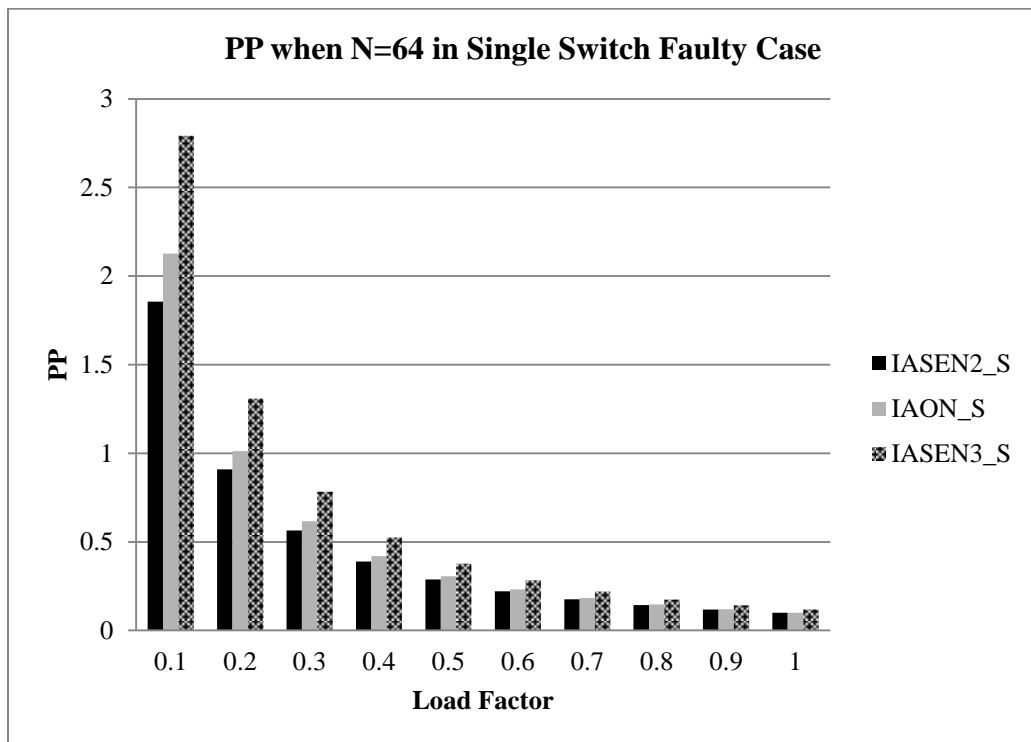


Figure 4.67. Comparison of PP for IASen-2, IAON and IASen-3 when N=64 in Single Switch Faulty Condition.

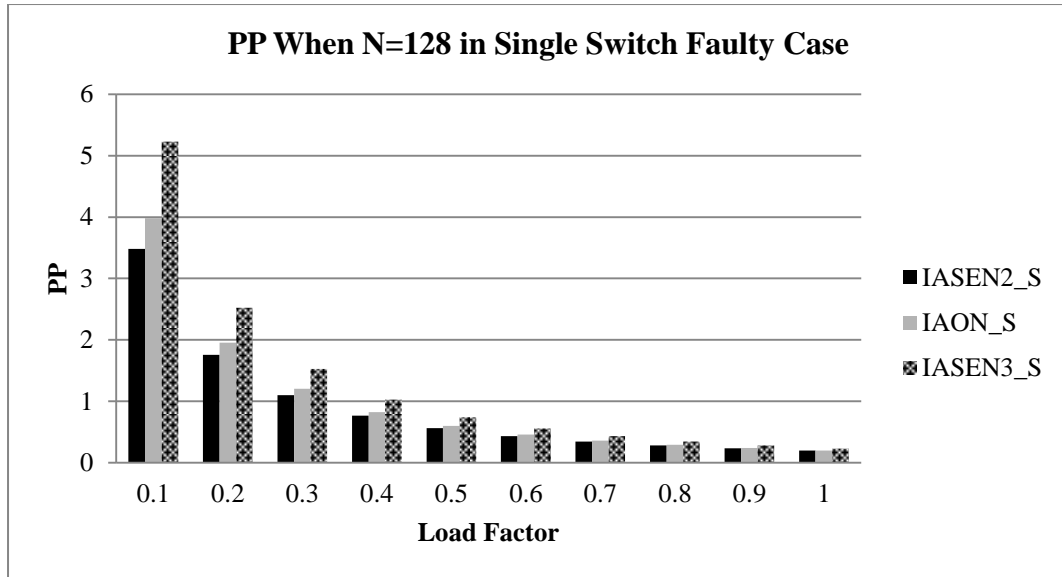


Figure 4.68. Comparison of PP for IASSEN-2, IAON and IASSEN-3 when N=64 in Single Switch Faulty Condition.

Here the author has shown the simulation results of IASSEN-2, IAON and IASSEN-3 (from figure 4.40 to 4.68). Further he calculated the average PA which is as follows:

Network	PA
IASSEN-2	0.564
IAON	0.588
IASSEN-3	0.714

Table 4.6

The table 4.4 clearly shows that the PA of IASSEN-2 and IAON are 56.4% and 58.8% whereas PA of IASSEN-3 is 71.4%.

It means IASSEN-3 is 15% better than IASSEN-2 and 12.6% better than IAON in terms of PA. Further, it is observed that performance of a network is primarily depending on the bandwidth because most of the factors of performance (e.g. TP, PU, PP) are calculated on the basis of bandwidth.

Therefore, the author has calculated the average bandwidth which is follows:

N	IASEN-2	IAON	IASEN-3
16	9.032	9.420	11.438
32	18.065	18.840	22.877
64	36.130	37.680	45.754
128	72.261	75.360	91.508

Table 4.7

From table 4.5, it is observed that when size of network is doubled then value of bandwidth also gets doubled. So,

For IASEN-2=>

$$\left(\frac{9.032}{16}\right) \times 100 = 56.4\%$$

Here the size of network is 16 hence the author has taken the value 16 in denominator and multiplied by 100 to calculate the percentage.

For IAON=>

$$\left(\frac{9.420}{16}\right) \times 100 = 58.8\%$$

For IASEN-3=>

$$\left(\frac{11.438}{16}\right) \times 100 = 71.4\%$$

Again, it is observed that IASEN-3 is 15% better than IASEN-2 and 12.6% better than IAON.

4.5.24 Throughput for IAON in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions

In figure 4.69, 4.70, 4.71 and 4.72, the author has shown the throughput of IAON for N=16, 32, 64 and 128 respectively. Here the throughput is calculated in non-faulty, single switch faulty and double switch faulty condition. Although, the throughput is decreasing in faulty conditions still it performs well in these conditions.

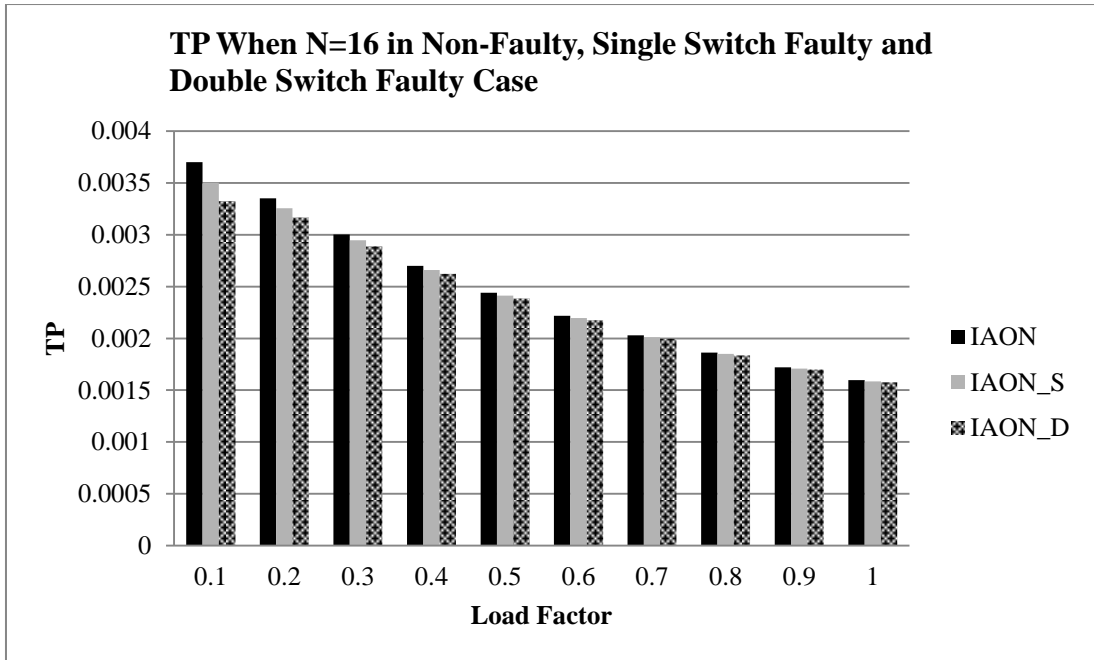


Figure 4.69. Comparison of TP for IAON when N=16 in Non-Faulty, Single Switch Faulty, and Double Switch Faulty Conditions.

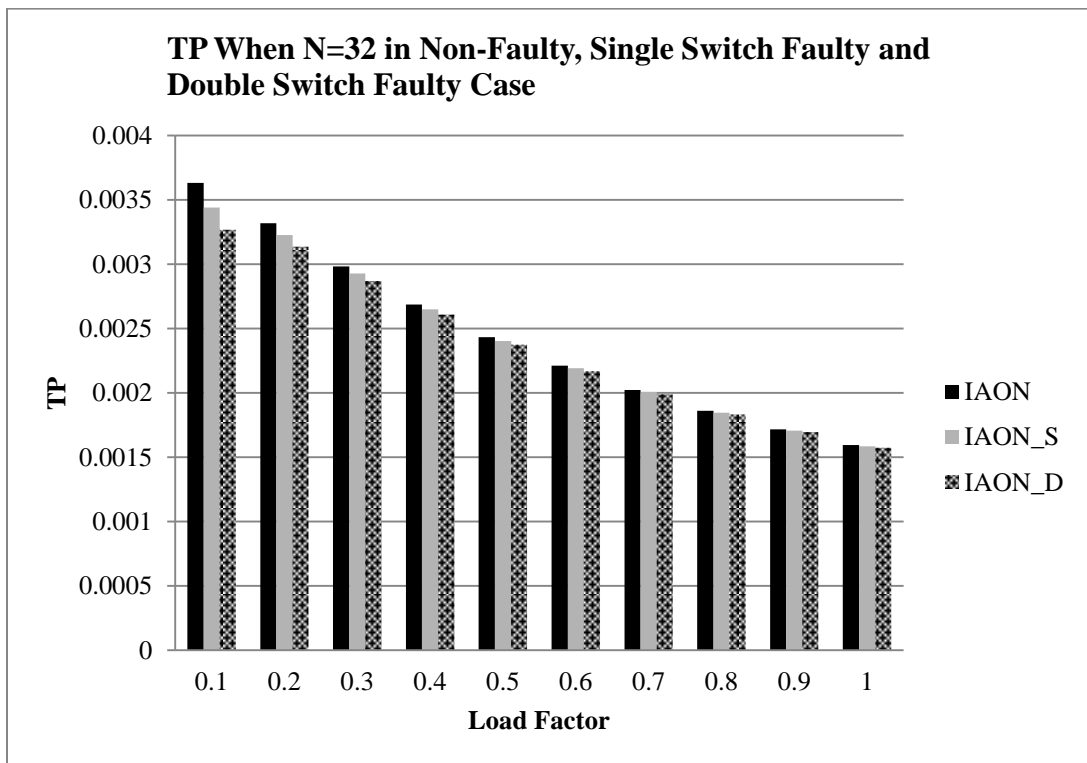


Figure 4.70. Comparison of TP for IAON when N=32, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

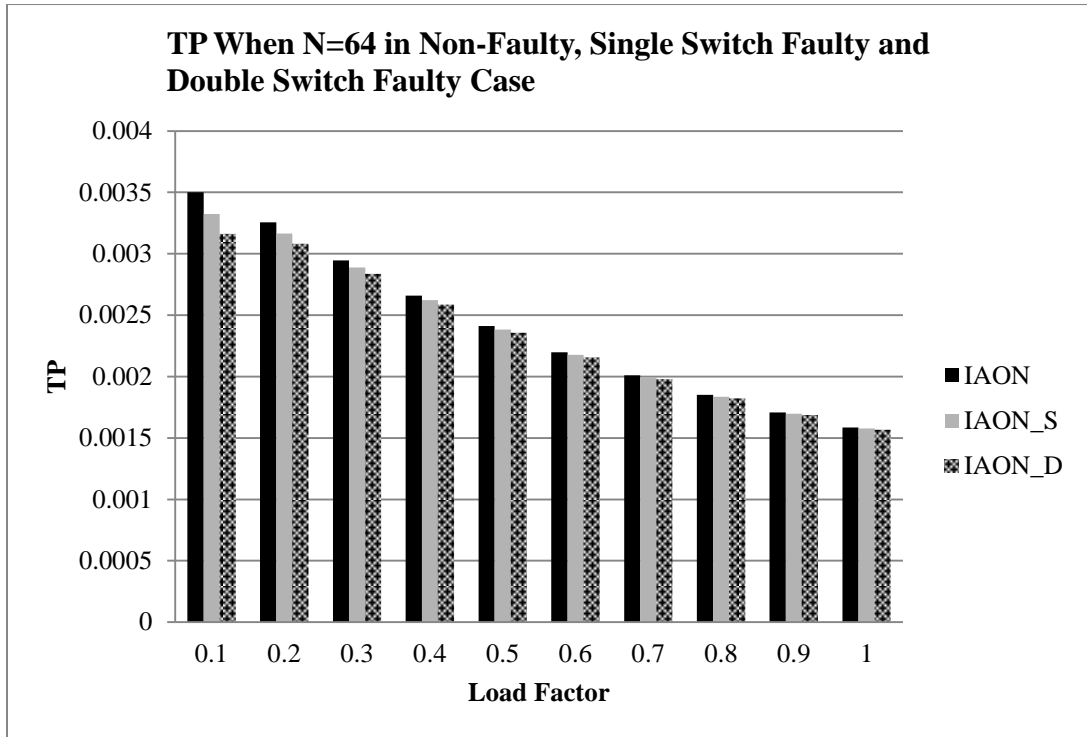


Figure 4.71. Comparison of TP for IAON when N=64, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

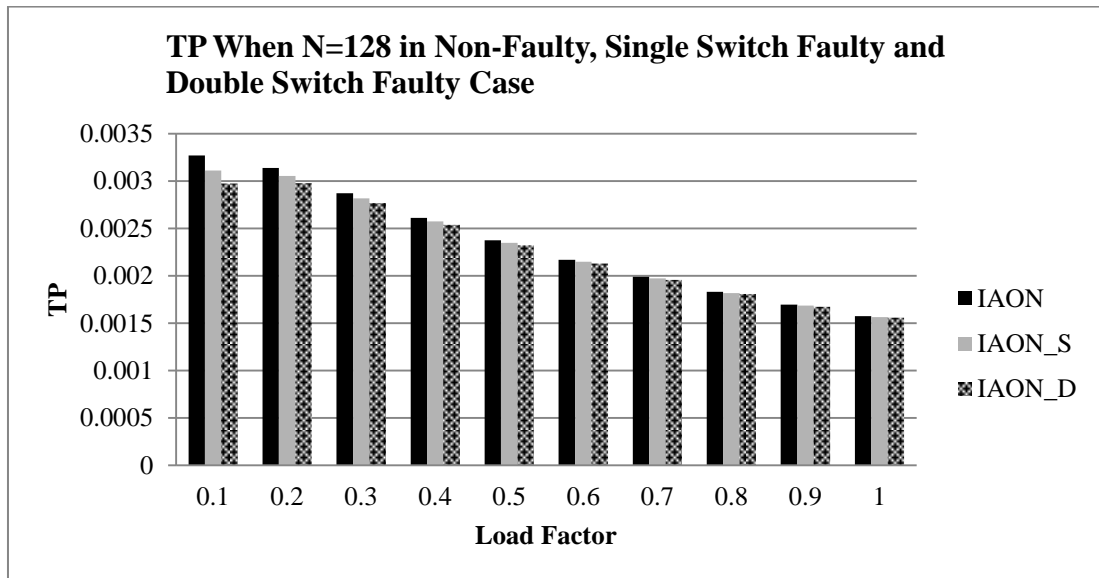


Figure 4.72. Comparison of TP for IAON when N=128, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

4.5.25 Processor Utilization for IAON in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions

In figure 4.73, 4.74, 4.75 and 4.76, the author has shown the processor utilization of IAON for N=16, 32, 64 and 128 respectively. Here the processor

utilization is calculated in non-faulty, single switch faulty and double switch faulty condition. Although, the processor utilization is decreasing in faulty conditions still it performs well in these conditions.

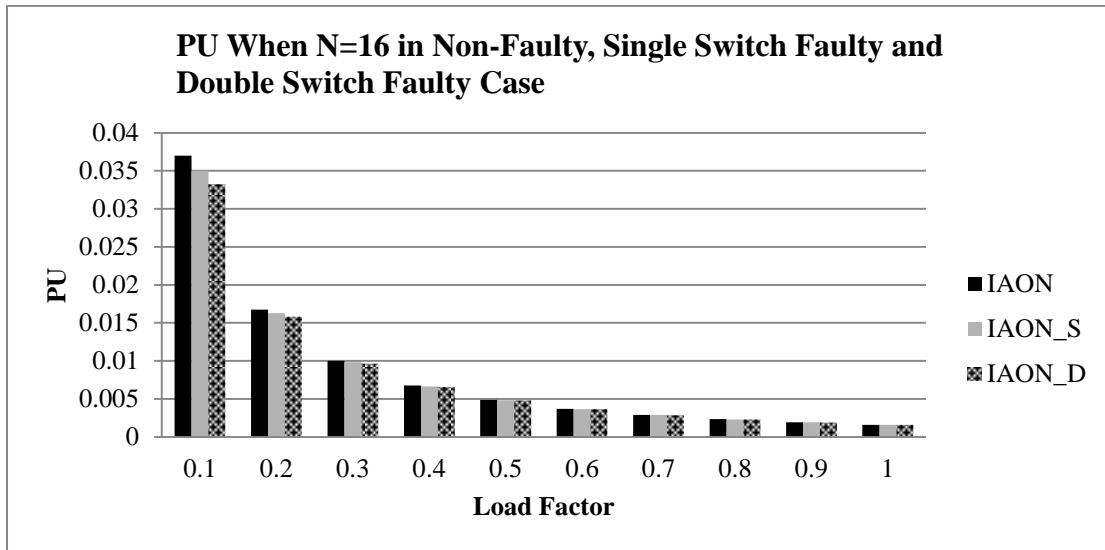


Figure 4.73. Comparison of PU for IAON when N=16, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

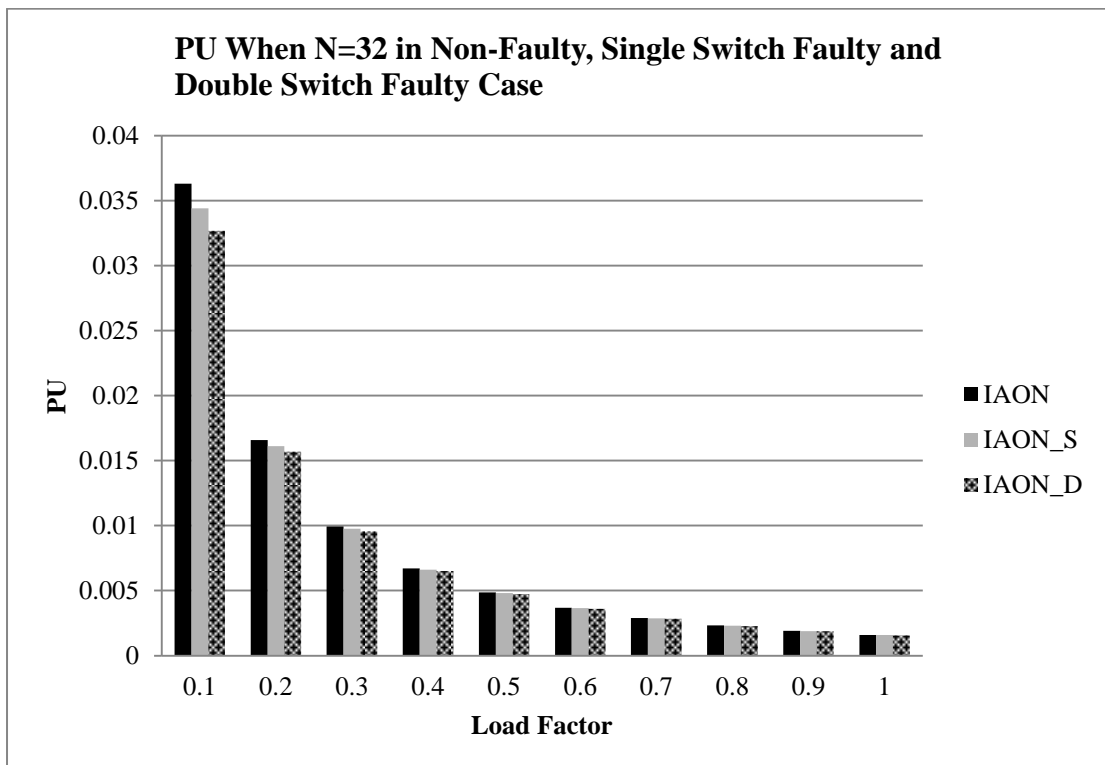


Figure 4.74. Comparison of PU for IAON when N=32, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

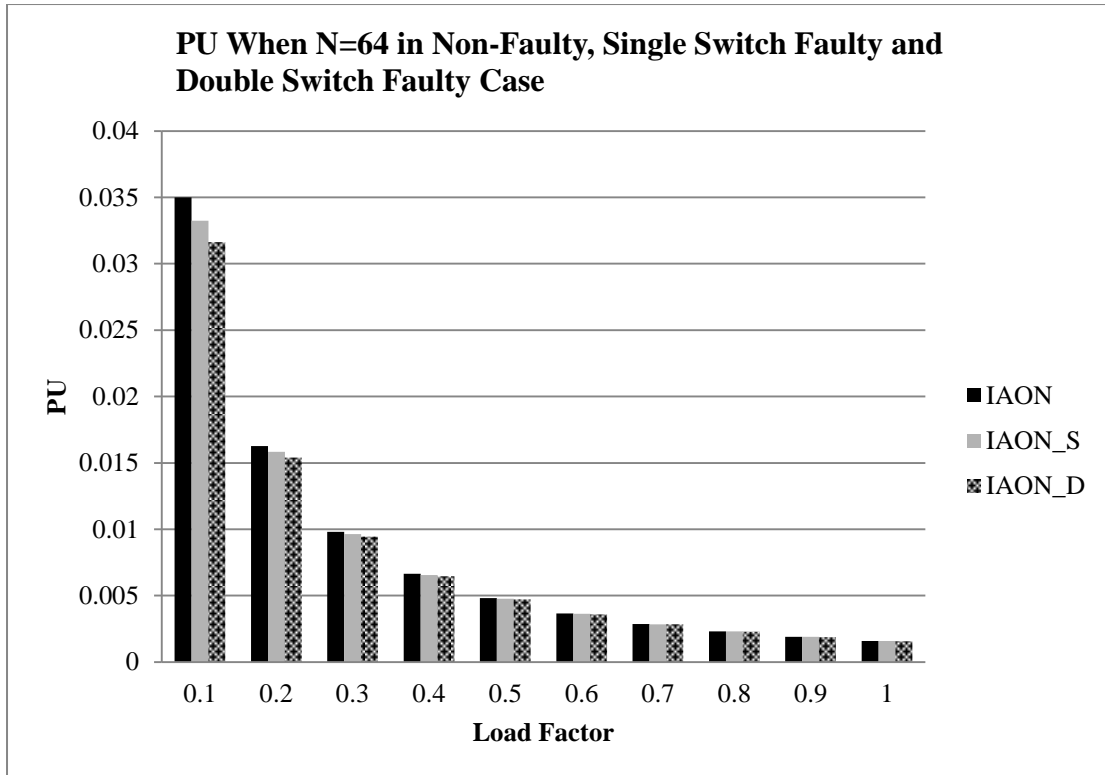


Figure 4.75. Comparison of PU for IAON when N=64, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

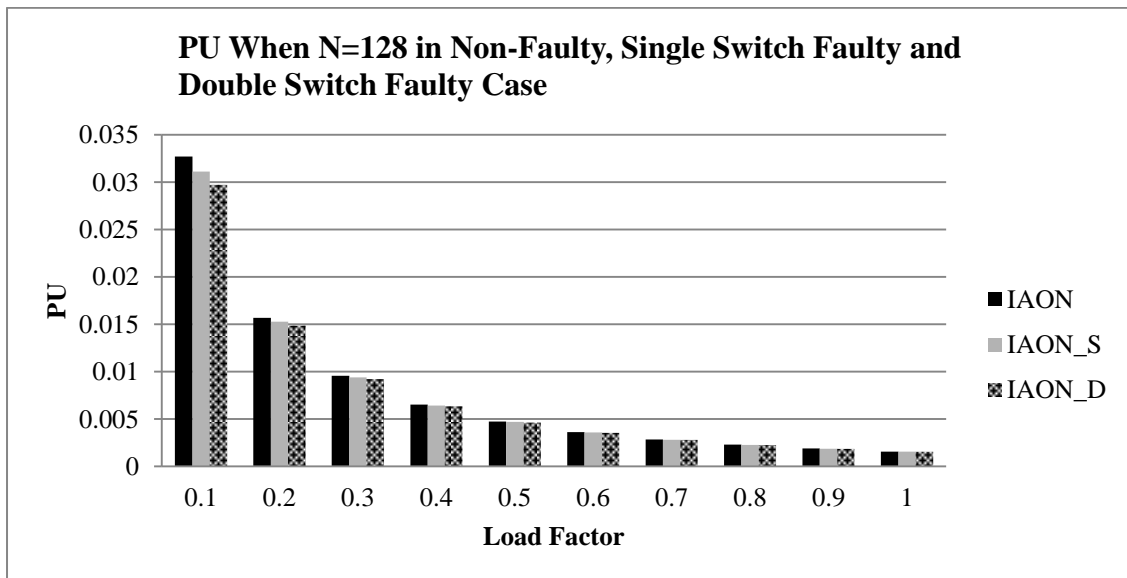


Figure 4.76. Comparison of PU for IAON when N=128, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

4.5.26 Processing Power for IAON in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions

In figure 4.77, 4.78, 4.79 and 4.80, the author has shown the processing power of IAON for N=16, 32, 64 and 128 respectively. Here the processing power is calculated in non-faulty, single switch faulty and double switch faulty condition. Although, the processing power is decreasing in faulty conditions still it performs well in these conditions.

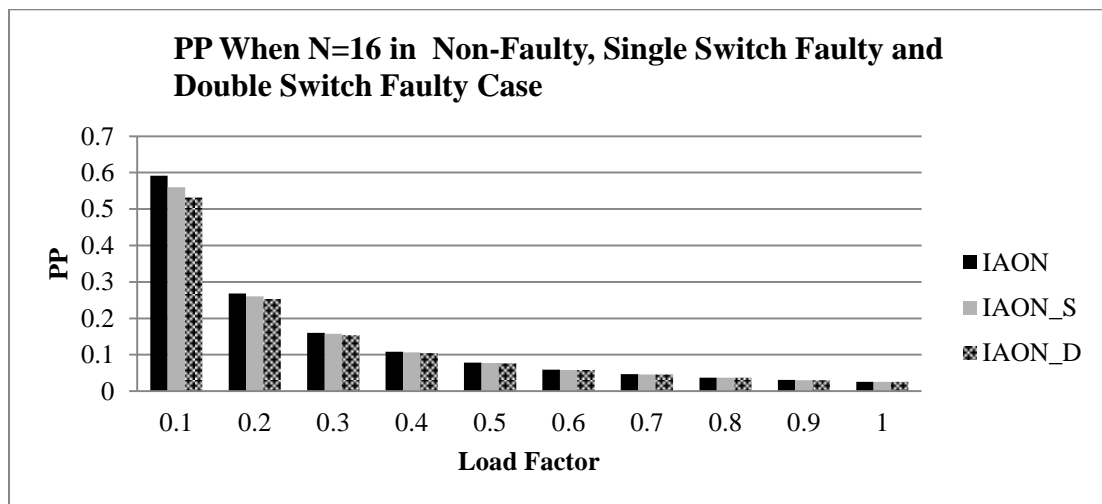


Figure 4.77. Comparison of PP for IAON when N=16, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

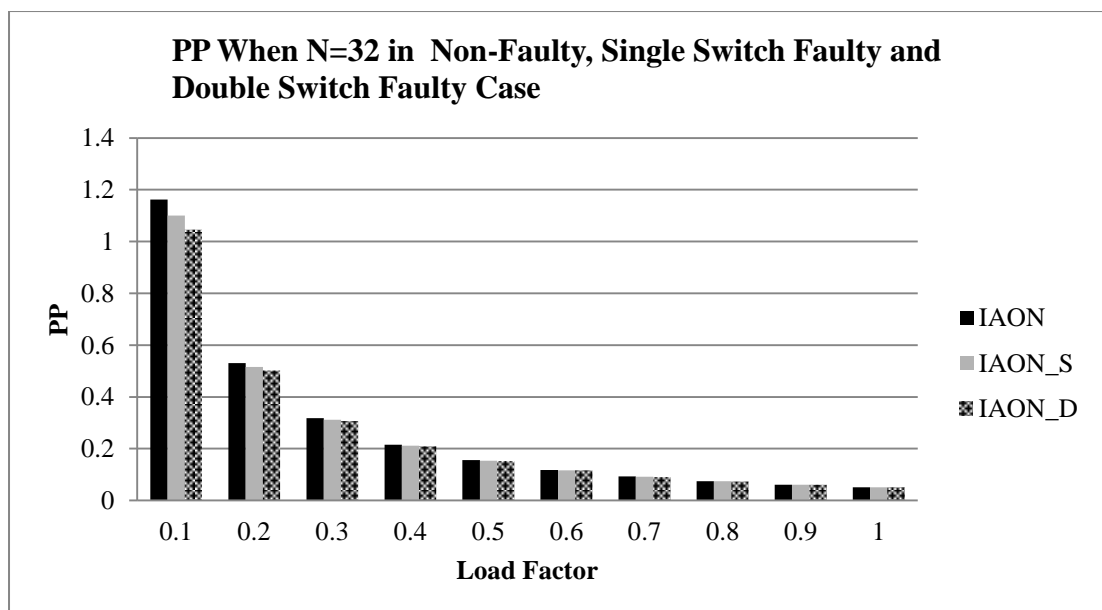


Figure 4.78. Comparison of PP for IAON when N=32, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

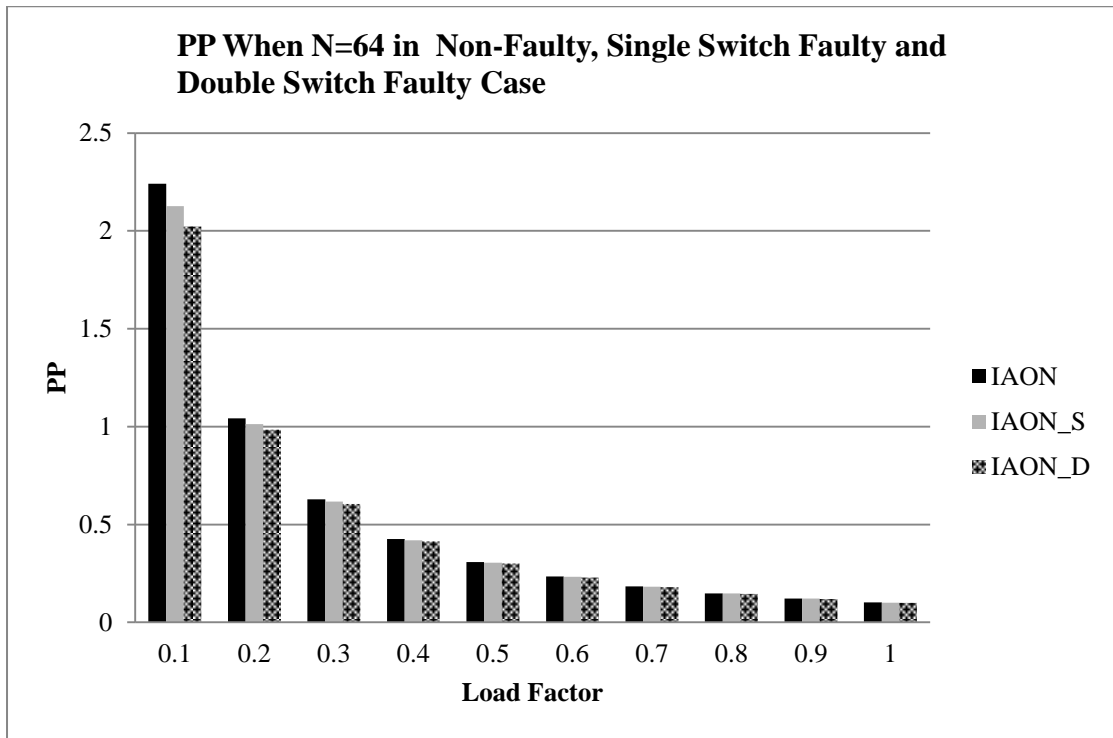


Figure 4.79. Comparison of PP for IAON when N=64, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

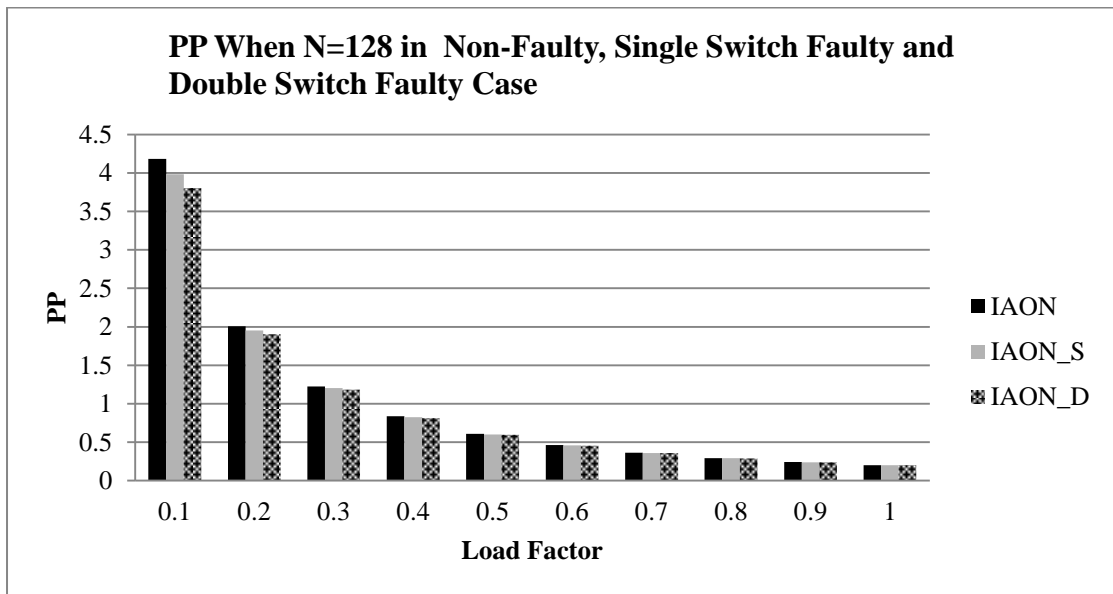


Figure 4.80. Comparison of PP for IAON when N=128, in Non-Faulty, Single Switch Faulty and Double Switch Faulty Conditions.

4.6 Conclusion and Future Scope of the Work

In this chapter, the author has proposed two new fault tolerant network models named as irregular advance omega network and irregular augmented shuffle exchange network-3. The explanation of routing algorithms and theorems of IAON and IASEN-3 illustrates that all the proposed networks are good in terms of fault sustainability and performance. The performance of all the networks is based on bandwidth, the probability of acceptance, throughput, processor utilization and processing power.

IAON and IASEN-3 gives better performance than the MALN and IASEN-2 in non-faulty and single switch faulty conditions. Further, it is observed that IASEN-3 has better performance than IAON. However, IAON has double switch faulty tolerability and IASEN-3 has single switch fault tolerability. Performance of IAON is also calculated in double switch faulty case. It has been observed that in faulty conditions, the performance of each network is degraded. However, all the proposed MINs are still performing well in faulty situations than the earlier proposed MINs. In future, the author wants to apply new connection pattern on IAON and IASEN-3 so that both the networks can work effectively in crucial faulty situations.

CHAPTER 5

MINIMIZING CROSSTALK IN MINS

5.1 Introduction

Parallel computing applications need high speed and reliability. Multi-stage interconnection networks (MINS) accomplish this necessitates [1-6]. Generally, it has N inputs and N outputs. N is the size of the network. It comes in the category of dynamic interconnection networks. Crosstalk [42] is the major shortcoming of MINS. It limits the size of the network and reduces the signal to noise ratio. Therefore, it affects the data transmission process in a negative manner. It is the basic reason of distortion of data packets which are transmitted through the network and hence, crosstalk degrades the efficiency of a MIN. In this chapter, the main focus of the author is to reduce the crosstalk and increase the efficiency of MINS. Here, crosstalk refers to switch crosstalk problem. It can be defined as follows:

“Crosstalk is a situation when two paths sharing a directional coupler experience unwanted coupling from one path to the other. Switch crosstalk is the most significant factor which reduces clarity of an optical signal, limits the size of a network and leads to error rate degradation [42]”.

Further, the author has proposed an algorithm named as an address selection algorithm (ASA) and a new MIN named as destination based modified omega network (DBMON) in order to reduce the crosstalk problem. ASA is applied on original omega network and Clos network. For message transmission through DBMON, the author has designed its routing algorithm named as destination based scheduling algorithm (DBSA). DBSA schedules and routes the data packets in such a way that the DBMON always remains crosstalk free for network size.

In table 5.1, various symbols and their meaning are defined which are used throughout the chapter.

Symbol	Meaning of Symbol
SA	Source Address
DA	Destination Address
Address	SA and its corresponding DA
NoP	Number of Passes
USW	Upper straight way
LUS	Lower to upper straight way
LSW	Lower straight way
ULS	Upper to lower straight way

Table 5.1. Symbol Table.

5.2 Related Work

In literature, various approaches have been presented against the crosstalk problem, e.g. space domain approach (SDA), time domain approach (TDA). SDA [23-29] is an expensive approach as it converts a network of size $N \times N$ into $2N \times 2N$ to minimize the crosstalk. Further, in TDA crosstalk is considered as a conflict. In this approach, data packets are passed in different time slots to avoid the crosstalk. The author has considered the TDA in his research work.

Various methods are based on TDA [23-29] e.g. window method, improved window method, heuristic routing algorithms. All these methods are explained in chapter 2. Further, a new MIN named as crosstalk free modified omega network (CFMON) has been proposed by Sonia Kaur and Neeraj Shukla. The detailed description of CFMON is given in the next subsection.

5.2.1 Crosstalk Free Modified Omega Network

Omega and clos are the two existing MINs which were taken as the base network to construct the CFMON. Figure 5.1 shows the structure of 8×8 CFMON [28].

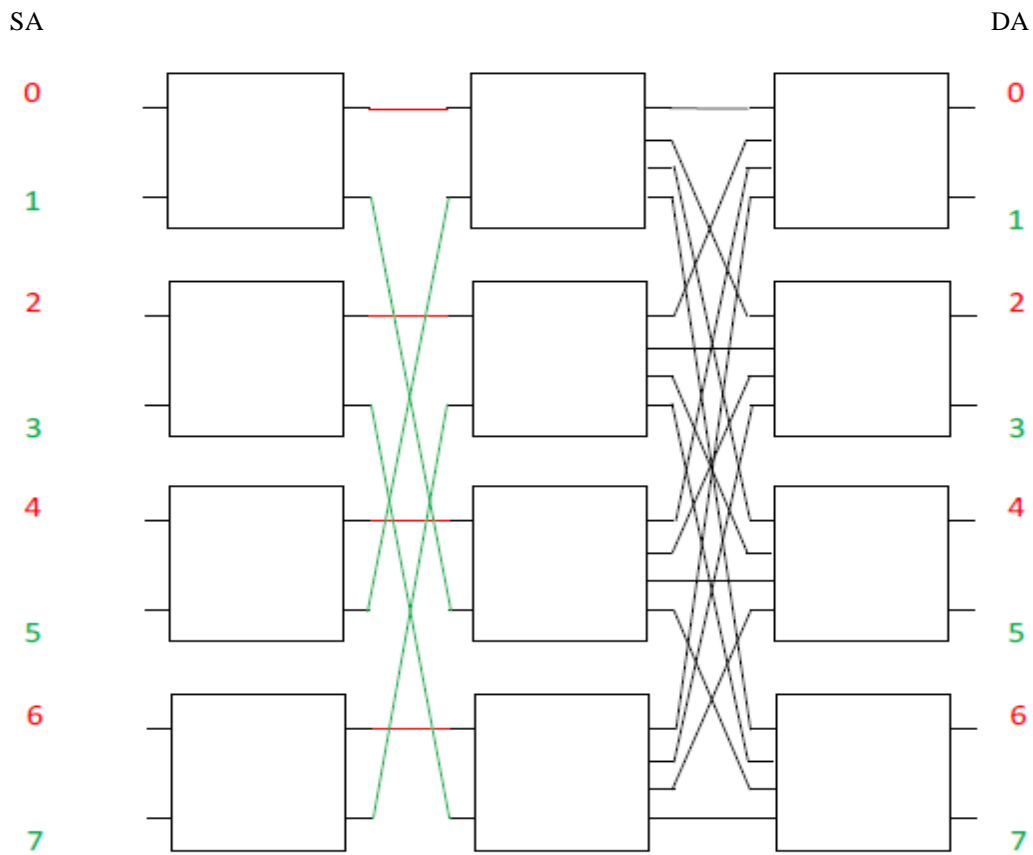


Figure 5.1. 8x8 CFMON.

It has $N=8$ inputs and $N=8$ outputs. Here N is the size of the network. The connectivity between the first and middle stage is same as in omega network. The connectivity between the middle and last stage is same as in Clos network. In CFMON [28], the routing of data packets is based on its routing algorithm. According to the algorithm, the source address (SA) which is shown by green color in figure 5.1 will send the data packets to their given destination addresses (DAs) in the first pass. The rest of the SAs will send the data packets to their given DAs in the second pass.

5.3 Address Selection Algorithm

This algorithm deals with the scheduling of messages based on the SAs and their DAs. Figure 5.2 shows the ASA. Selection of specific SAs and their DAs for data transmission depends on some mathematical steps which are given in the algorithm. The addresses which can create conflict in the network are scheduled for different time slots so that they can send the data packets separately or without crosstalk.

Address Selection Algorithm (ASA)

1. Get the source and destination address sequentially.
2. Make a combination matrix of the source and corresponding destination address.
3. Transform the matrix. A complete set of rows are thus formed $r_0, r_1 \dots r_n$, where n = total number of bits in source and destination address.
4. Select the middle four rows.
5. Pair the obtained rows in set of two.
6. Add the corresponding bits of each pair.
7. Subtract the result of the first set from the second set.
8. If (result ≤ 0)
 Then take a corresponding address and transmit them in current pass and go to step 9.
 Else
 Store the address in remaining_address.
9. If (Conflict)
 Then transmit the address with higher magnitude of the conflicting address pair and add the lower magnitude address to the remaining_address.
10. Transmit the remaining_address.
11. End.

Figure 5.2. Address Selection Algorithm.

The addresses which can create conflict in the network are scheduled for different time slots so that they can send the data packets separately or without crosstalk. The term “addresses” refers to the given SAs and their DAs.

On the large size of the network, the ASA cannot reduce the crosstalk. It is the limitation of ASA. Further, to understand the ASA the author has considered the following example 1.

Example 1: It is supposed that the SAs and their corresponding DAs for network size (N) = 8 are as follows:

SA	DA
000	100
001	011
010	101
011	110
100	010
101	001
110	000
111	111

Table 5.2. SAs and DAs.

Algorithmic Step1:

Obtain the source and corresponding destination as given in table 5.2.

Algorithmic Step2:

According to the algorithm, obtain the combination matrix as shown in figure 5.3.

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Figure 5.3. Combination Matrix.

Algorithmic Step3:

In this step, transpose of the combination matrix will be obtained. It is shown in figure 5.4.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Figure 5.4. Transpose of Matrix.

Algorithmic Step4:

In this step, middle 4 rows will be selected as given in figure 5.5.

$$\begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Figure 5.5. Selection of Middle 4 Rows.

Algorithmic Step5:

In this step, pairing of rows will be performed as shown in figure 5.3.

r ₁	r ₂	r ₃	r ₄
0	0	1	0
0	1	0	1
1	0	1	0
1	1	1	1
0	0	0	1
0	1	0	0
1	0	0	0
1	1	1	1

Table 5.3. Pairing of Rows.

Algorithmic Step6:

Do the addition of r1 and r2. Similarly add r3 and r4 as given in table 5.4.

r ₁ +r ₂	r ₃ +r ₄
0	1
1	1
1	1
2	2
0	1
1	0
1	0
2	2

Table 5.4. Addition of Rows.

Algorithmic Step7:

Now, subtracts the r₁+r₂ from r₃+r₄. It is given in table 5.5.

$(r_1+r_2) - (r_3+r_4)$
-1
0
0
0
-1
1
1
0

Table 5.5. Subtraction of Rows.

Algorithmic Step8:

As given in the algorithm, the addresses which have got 0 or -1 in table 5.5 will be transmitted in the first pass.

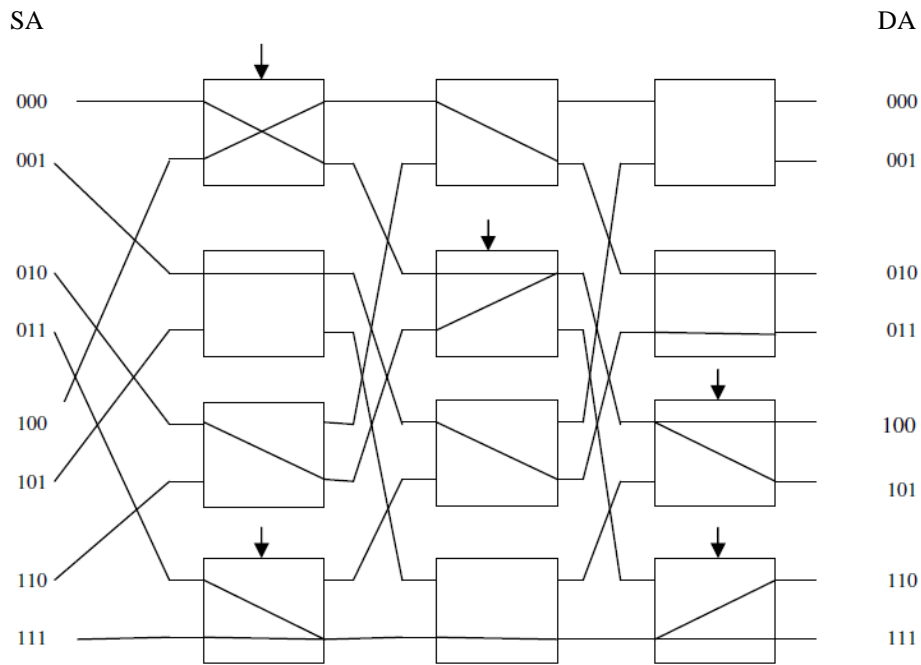


Figure 5.6. Transmission of Selected Addresses.

The SAs 000, 001, 010, 011, 100 and 111 will transmit the data packets as shown in figure 5.6. The rest of the SA and their DAs will be stored in a variable named as remaining_address.

Algorithmic Step9:

In figure 5.6, the conflict is shown by black arrows. After selecting the conflicting SAs, it is required to separate the SAs from the conflicting SAs group based on their magnitude. If any SA has higher magnitude from its conflicting SA than it will be passed in the next pass as shown in figure 5.7. The SAs which has lower magnitude than the conflicting SAs will be stored in variable remaining_address.

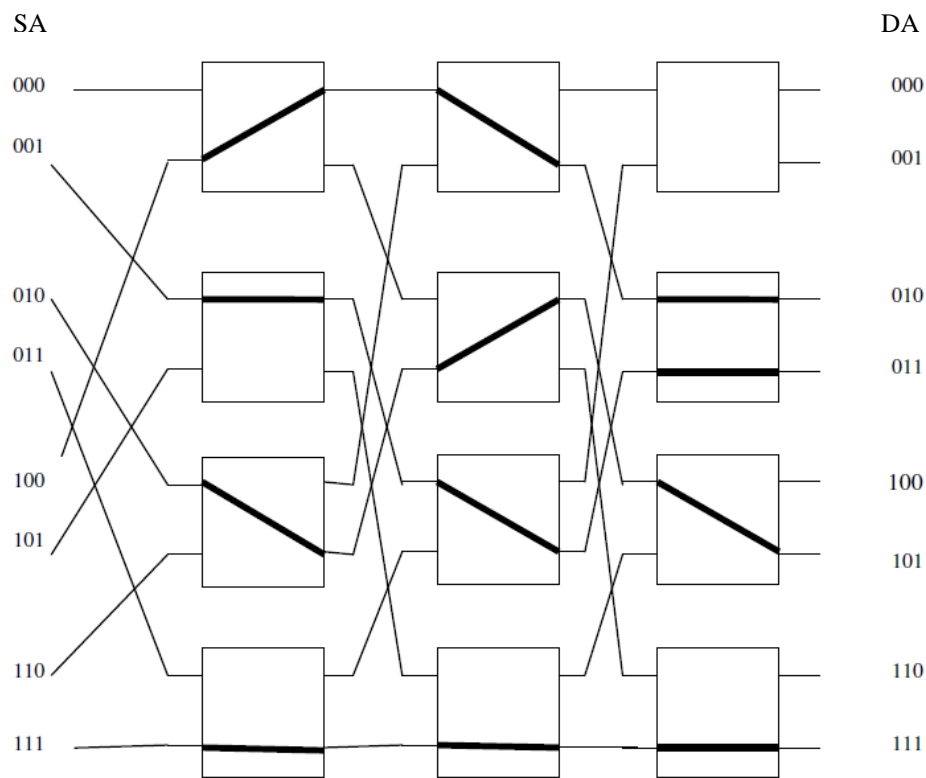


Figure 5.7. Transmission of SAs of High Magnitude.

In figure 5.7, it can be observed that the network has crosstalk in second SE of third stage.

Algorithmic Step10:

In this step, the SAs which are stored in remaining_address will transmit the data packets to their given DAs.

Algorithmic Step11: In this way, crosstalk is minimized by the ASA.

The algorithmic steps describe the process that how the ASA produces the crosstalk free routes. Actually, ASA filters the SAs and DAs based on certain mathematical calculation but it cannot be used in a generalized way because it takes more number of passes and time in case of the network of large sizes. Therefore, it is good for the network of small size. In the next subsection, it is applied to the Clos network.

5.3.1 Applying ASA on Clos Network

Clos Network (CN) [28, 52] is a multi-stage interconnection network. This is a nonblocking interconnection network. Generally, the Clos network has $n \times k$ switches in its first stage, $m \times m$ switches in its second stage and $k \times n$ switches in its final stage. In first stage, each switch has n inputs and k outputs, in the second stage, each switch has m inputs and m outputs and in the final stage each switch has k inputs and n outputs. There are $n \times m$ SAs and $n \times m$ DAs in CN.

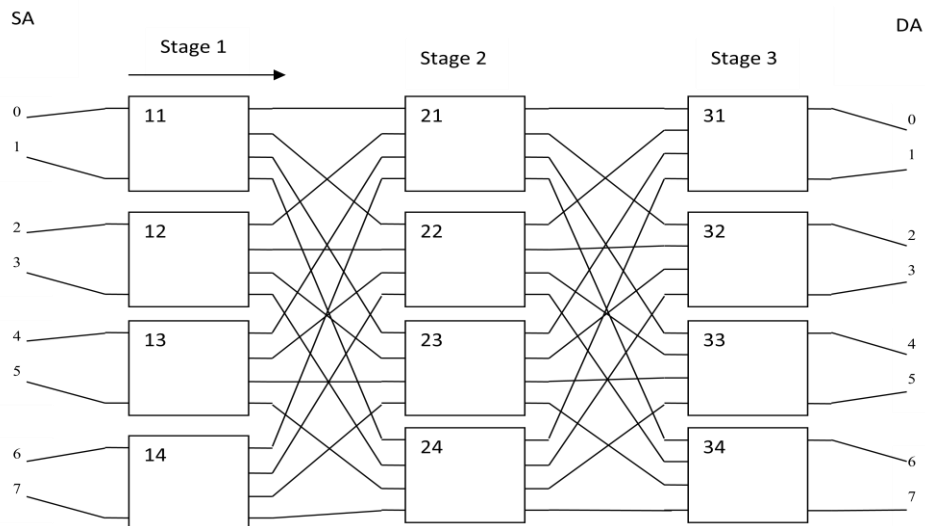


Figure 5.8. 8x8 Clos Network.

Further, k represents the total number of switches in the middle stage and the nonblocking condition for CN is $k \geq (2n-1)$. In figure 5.8, a 3-stage 8x8 CN has been shown. Figure 5.8 shows that the value of $n=2$, $k=4$, $m=4$ and therefore, it satisfies the nonblocking condition i.e. $4 \geq 3$. In this chapter, the author has given a unique number to every switch and therefore each switch is represented by a pq switch in a generalized way. Here p is the number of stage and q is the number of switches. The value of $p=1, 2$ or 3 and the value

of $q=1, 2, \dots, k$. Suppose, the value of p is 3 and q is 2, it indicates the second switch which exists in the third stage in figure 5.8. Similarly, one can identify the other switches in the network. Further, the author has transmitted all the SAs to their DAs in a single pass. After passing all the SAs and their DAs in a single pass, it is observed that the entire network has the crosstalk problem. The SE which is represented by dotted lines will be considered as conflicted SE. Figure 5.9 shows the conflicted SEs in the network.

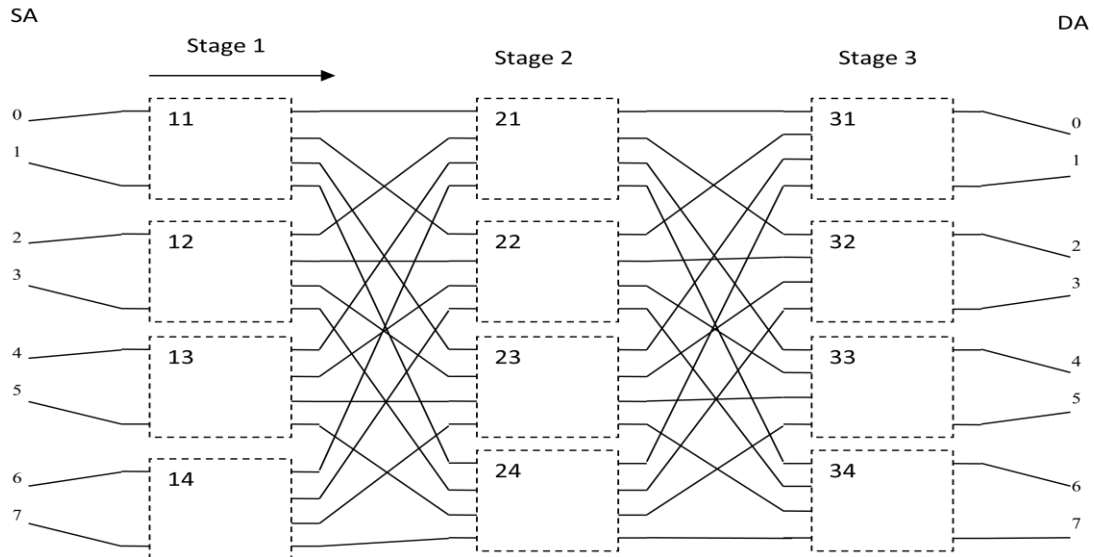


Figure 5.9. 8×8 Clos Network with Crosstalk.

To apply the ASA on clos network, the author has taken the following example:

Example: 2 It is supposed that the SAs and their DAs for $N = 8$ are as follows:

SA	DA
000	111
001	110
010	101
011	100
100	011
101	010
110	001
111	000

Table 5.6. SAs and DAs.

Algorithmic Step1: Get the source and destination address sequentially and obtained a combination matrix of source and corresponding destination address, e.g. the source address is 000 and its corresponding destination address is 111 and therefore they will become 000111. Similarly the other combination can be obtained. Figure 5.10 shows the combination matrix of 8×8 CN.

$$\begin{bmatrix} 000111 \\ 001110 \\ 010101 \\ 011100 \\ 100011 \\ 101010 \\ 110001 \\ 111000 \end{bmatrix}$$

Figure 5.10. Combination Matrix.

Algorithmic Step2: In this step obtained the transpose of the combination matrix. Figure 5.11 shows the transpose matrix of 8×8 CN.

$$\begin{bmatrix} 00001111 \\ 00110011 \\ 01010101 \\ 11110000 \\ 11001100 \\ 10101010 \end{bmatrix}$$

Figure 5.11. Transpose Matrix.

Algorithmic Step3: Selection of the middle 4 rows is performed in this step. Figure 5.12 shows the selected rows of 8×8 CN.

$$\begin{bmatrix} 00110011 \\ 01010101 \\ 11110000 \\ 11001100 \end{bmatrix}$$

Figure 5.12. Selected Rows.

Algorithmic Step4: Pairing is performed in this section. It is shown in table 5.7.

Row1	Row2	Row3	Row4
0	0	1	1
0	1	1	1
1	0	1	0
1	1	1	0
0	0	0	1
0	1	0	1
1	0	0	0
1	1	0	0

Table 5.7. Pairing of Rows.

Algorithmic Step5: Addition of corresponding bits is performed in this step. It is shown in table 5.8.

Row1+Row2	Row3+Row4
0	2
1	2
1	1
2	1
0	1
1	1
1	0
2	0

Table 5.8. Addition of Corresponding Bits.

Algorithmic Step6: Subtraction is performed in this step. It is shown in table 5.9.

(Row1+Row2) -(Row3+Row4)
-2
-1
0
1
-1
0
1
2

Table 5.9. Subtraction.

Algorithmic Step7: In this step it is required to check the result, if it is 0 or less than 0 then select its corresponding SA for the first pass or store the address in the remaining_address variable for the next pass and therefore the SAs 000,001,010,100 and 101 will be sent in the first pass. Figure 5.13 shows the first pass through the clos network. The SEs which is shown by dotted lines in figure 5.12 is having crosstalk. The rest of the SEs in the network is crosstalk free.

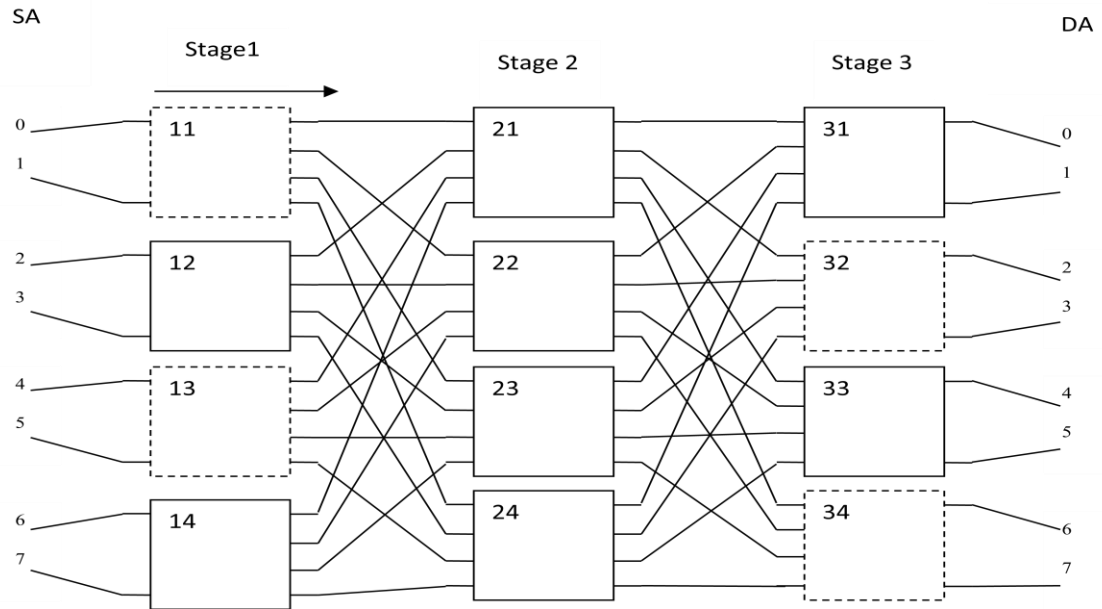


Figure 5.13. First Pass of Selected SAs and their DAs.

Algorithmic Step8: After the first pass, check the conflicts of the first stage and select the source address which is high in magnitude from the conflicting address pair for the next pass and put the other source address in the remaining_address variable. Here the conflicting addresses are 0, 1 and 4, 5 and therefore, the addresses 1 and 5 will be selected for the second pass and the rest of the addresses will be stored in the remaining_address variable for the next pass.

Algorithmic Step9: In this step, transmission of the SAs and their DAs will be performed which are stored in remaining_address. This variable has the SAs which are eliminated before the first pass and the SAs which are eliminated after the first pass.

Algorithmic Step10: The desired goal is achieved. Further, it is analyzed that the number of conflicted switches in the CN is less after the first pass of the ASA. The comparison between the number of conflicted switches in figure 5.9 and figure 5.13 shows that ASA has reduced the conflicted SEs in the network. Therefore, it can be said that the proposed algorithm can also solve the crosstalk problem effectively in the nonblocking networks and it can provide the conflict free routes in the network.

5.4 Destination Based Modified Omega Network

The explanations about the destination based modified omega network (DBMON) are given in the next subsections.

5.4.1 Structure of DBMON

DBMON is constructed on the basis of omega network [16-19]. However, the connectivity amid the stages is different. It is shown by dotted lines in figure 5.14. Here SA and DA are the source and destination address respectively. This network has $m = \log_2 N$ stages and there are $2^{(m-1)}$ SEs in each stage. In this way, DBMON contains $TSE = (\log_2 N \times 2^{m-1})$ SEs. Here TSE represents the total number of SEs. In DBMON, PS_r represents a SE. PS_r shows that P^{th} SE exists in the r^{th} stage of DBMON. Here $P = 1, 2, \dots, 2^{m-1}$ and $r = 1, 2, \dots, m$.

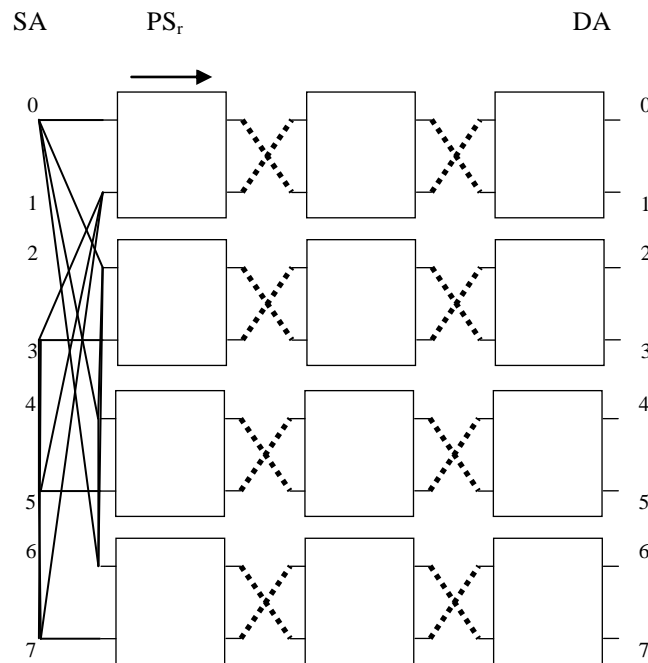


Figure 5.14. 8x8 DBMON.

If the given value is $2S_3$ then it indicates the second SE of the third stage in the network. Similarly, the other SEs can be identified.

5.4.2 Internal Routing in DBMON

The routing process of DBMON is based on destination tag routing. The route of data packets depends on their DAs. When any data packet reaches on the

upper input link of a SE and the given bit is 0 then the data packet will follow the upper straight way (USW) as shown in figure 5.15 (a). When any data packet reaches on the lower input link of a SE and the given bit is 0 then the data packet will follow the lower to upper straight way (LUS) as shown in figure 5.15 (b). Further, the vice versa is true for the bit 1 and it is given in figure 5.16 (a) and 5.16 (b). LSW and ULS are the lower straight way and upper to lower straight way respectively.

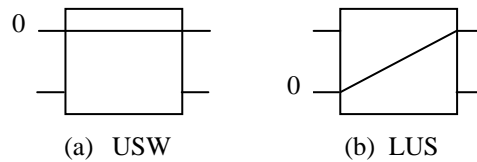


Figure 5.15. Routing of Bit '0' in DBMON.

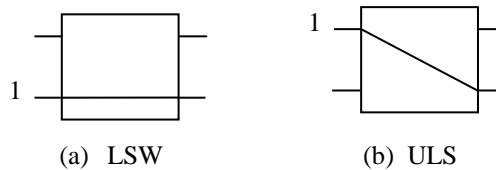


Figure 5.16. Routing of Bit '1' in DBMON.

5.4.3 Destination Based Scheduling Algorithm

The DBSA algorithm is the generalized form of the ASA algorithm. ASA was performing well for the networks of small sizes. To have a generalized solution against the crosstalk, the author has proposed the DBMON and its scheduling algorithm named as a destination based scheduling algorithm (DBSA). DBSA performs well on DBMON for every network size. It is shown in figure 5.17. There are 3 user defined functions in DBSA. The function INSERT-BEGINNING (head, SA, DA) shows that it will take SAs and their DAs as an input. Scheduling of SAs and their DAs is performed by the function EVEN-SELECTION (head) and ODD-SELECTION (head) for first and second pass respectively. The functions EVEN-SELECTION (head) and ODD-SELECTION (head) take the SAs which has even and odd destinations respectively. The complexity of the DBSA is $O(n)$.

Algorithm_DBSA

Input: N, SAs and DAs

Output: Scheduled SAs and DAs for first and second pass

Begin

INSERT-BEGINNING (head, SA, DA)

1. ptr.sinfo = SA
2. ptr.dinfo = DA
3. if LIST IS EMPTY
4. ptr.next = NULL
5. head = ptr
6. else
7. ptr.next = head
8. head = ptr

End INSERT-BEGINNING (head, SA, DA)

EVEN-SELECTION(head)

1. while head.next!=NULL
 2. if head.dinfo%2= 0
 3. head.sinfo
 4. head.dinfo
 5. End if
 6. head=head.next
- End while

End EVEN-SELECTION (head)

ODD-SELECTION(head)

1. while head!=NULL
 2. if head.dinfo%2!=0
 5. head.sinfo
 6. head.dinfo
 7. End if
 8. head = head.next
- End while

End ODD-SELECTION(head)

End

Figure 5.17. DBSA.

5.4.4 Path Information Algorithm

After obtaining the schedule of given SA and their DA, it required to know the route of these addresses. Path information algorithm (PIA) enlightens the route for each SA so that it can send the data packets on the given DA. The functions EVEN-DESTINATION (N) and ODD-DESTINATION (N) give the routing information of even and odd DAs respectively. It is shown in figure 5.18.

```
Algorithm_PIA
Input: N
Output: Route Information
Begin
EVEN-DESTINATION (N)
1.  for j = 0 to N && k = 0 to N/2
2.      DA [j] = j
3.      SE [k] = k+1
      End for
End EVEN-DESTINATION (N)
ODD-DESTINATION (N)
1.  for m = 1 to N && n=0 to N/2
2.      DA [m] = m
3.      SE [n] = n+1
      End for
End ODD-DESTINATION (N)
End
```

Figure 5.18. PIA.

Complexity of PIA is $O(n)$. The sum of the complexities of DBSA and PIA is $2 O(n)$ or $O(n)$.

5.4.5 Performance Evaluation Parameters

In this section the author has presented the factors of performance in order to compare the performances of CFMON and DBMON.

5.4.5.1 Maximum Number of Passes

Any network takes a particular number of passes (NOP) [28] in order to avoid the crosstalk for the given SAs and their DAs. The network, which takes small NOP will be counted as a good network.

5.4.5.2 Transmission Time (t)

It is the time which is taken by the data packets from the given SA to given DA through the given network [1-6].

5.4.5.3 Maximum Transmission Time (tt)

In this chapter, the calculation of maximum transmission time is based on the following parameters:

- Maximum NOP,
- Transmission Time

5.4.6 Performance Comparison of CFMON and DBMON

Comparison of the performances of both networks is explained by the following example:

Example 3: It is supposed that the SAs and their DAs for $N = 8$ are as follows:

SA	DA
0	7
1	3
2	6
3	2
4	1
5	5
6	0
7	4

Table 5.10 SAs and DAs.

Further, the given SAs and their DAs are passed through the CFMON. Before the transmission, the algorithm of CFMON is applied on the example 3 and the applied algorithm gives the following result for the first pass:

SA	DA
0	7
2	6
4	1
6	0

Table 5.11 First Pass of CFMON.

After observing the table 5.11, it can be said that the CFMON will face the crosstalk in 1st and 4th SEs of 3rd stage because of DAs 1, 0 and 7, 6 respectively. Therefore, CFMON required an extra pass for the data transmission without the crosstalk. The addresses which are selected for second pass are as follows:

SA	DA
1	3
3	2
5	5
7	4

Table 5.12 Second Pass of CFMON.

After observing the table 5.12, it can be said that the CFMON will face the crosstalk in 2nd and 3rd SEs of 3rd stage because of DAs 3, 2 and 5, 4 respectively. Therefore, CFMON required an extra pass for the data transmission without the crosstalk. After applying the algorithm of CFMON on given example, it can be said that CFMON requires maximum 4 passes for crosstalk free data transmission.

Further, the given SAs and their DAs are passed through the DBMON. Before the transmission, the DBSA is applied on the example 3 and it gives the following result for the first pass:

SA	DA	Allocated SE
2	6	4
3	2	2
6	0	1
7	4	3

Table 5.13 First Pass of DBMON.

After observing the table 5.13, it can be said that the route of each SA is unique because the allocated SE for data transmission is different. The allocated SEs is obtained from PIA.

Therefore, crosstalk will not occur in the network. Moreover, the selected SAs and DAs will be passed through the DBMON. It is shown in figure 5.19. The figure shows that the network is crosstalk free.

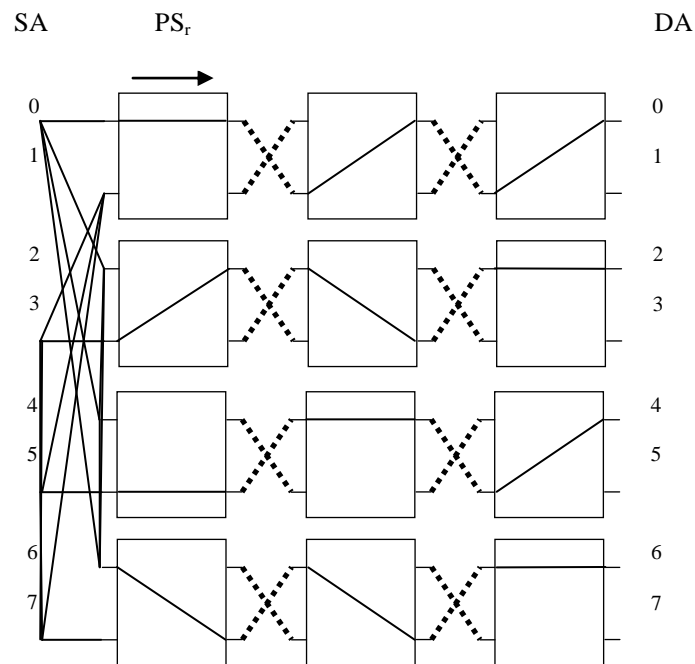


Figure 5.19. Crosstalk Free First Pass through DBMON.

Further, the selected SAs and DAs for second pass is as follows:

SA	DA	Allocated SE
0	7	4
1	3	2
4	1	1
5	5	3

Table 5.14 Second Pass of DBMON.

After observing the table 5.14, it can be said that the route of each SA is unique because the allocated SE for data transmission is different. Here, the allocated SEs is obtained from PIA. Therefore, crosstalk will not occur in the network. Further, DBMON requires only 2 passes for crosstalk free data transmission.

5.4.6.1 Comparison Based on Maximum Number of Passes

From example 3, it is clear that CFMON and DBMON require maximum 4 and 2 passes on every network size and produce the crosstalk free routes. It is shown in figure 5.20.

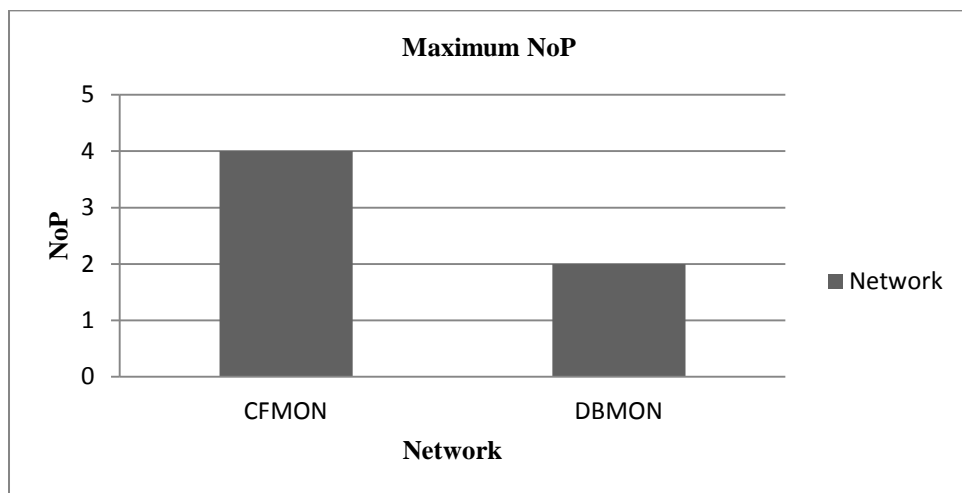


Figure 5.20. Comparison Based on Maximum NOP.

5.4.6.2 Comparison Based on Maximum Transmission Time

To obtain the maximum transmission time (TT), the author has assumed that the transmission time (t) of both networks is 1. Further, he has derived the following equation to get the TT:

$$\text{Network_TT} = (\text{Maximum NOP} \times t \times \frac{N}{8})$$

N is the size of the network. Here the minimum value of N is 8. Further, when the value of N will increase then the number of stages will also increase and therefore it will increase the value of t. Hence,

$$\text{CFMON_TT} = (4 \times t \times \frac{N}{8})$$

$$\text{DBMON_TT} = (2 \times t \times \frac{N}{8})$$

Further, CFMON_TT [28] and DBMON_TT are the maximum transmission time of CFMON and DBMON respectively. The graph of TT is shown in figure 5.21. It has the value in terms of t.

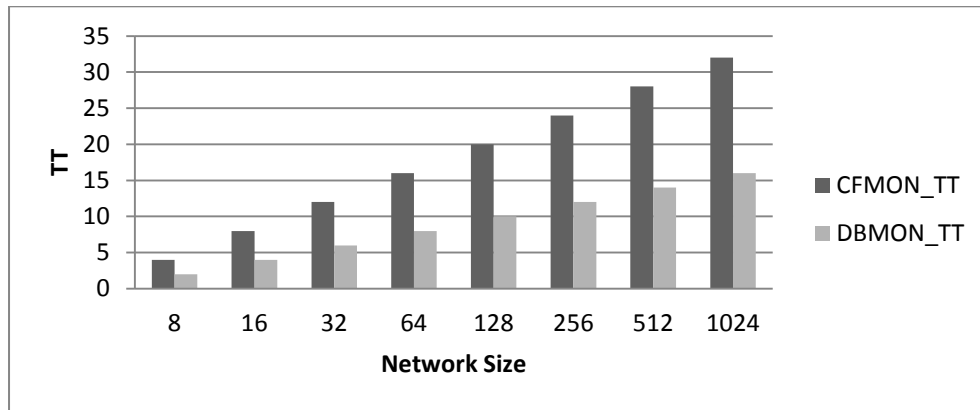


Figure 5.21. Comparison Based on Maximum Transmission Time.

The graph clearly shows that DBMON is better than CFMON against the crosstalk problem because it takes less number of passes, less transmission time than the CFMON and produce the crosstalk free routes.

5.4.6.3 Comparison Based on Throughput

The bandwidth (BW) of CFMON is calculated as follows:

$$p_1 = 1 - (1 - p_0/2)^2$$

$$p_2 = 1 - (1 - p_1/2)^2$$

$$p_3 = 1 - (1 - p_2/8)^2$$

$$p_4 = 1 - (1 - p_3/2)^8$$

$$\text{BW}_{\text{CFMON}} = (N \times p_4)$$

The bandwidth of DBMON is calculated as follows:

$$p_1 = 1 - (1 - p_0/2)^2$$

$$p_2 = 1 - (1 - p_1/2)^2$$

$$p_3 = 1 - (1 - p_2/2)^2$$

$$p_4 = 1 - (1 - p_3/2)^2$$

$$BW_{DBMON} = (N \times p_4)$$

The comparison in bandwidth (when N=16) is shown in table as follows:

p	CFMON	DBMON
0.1	1.4518	1.4513
0.2	2.6485	2.6452
0.3	3.6410	3.6318
0.4	4.4685	4.4505
0.5	5.1615	5.1323
0.6	5.7441	5.7020
0.7	6.2351	6.1791
0.8	6.6496	6.5794
0.9	6.9999	6.9154
1.0	7.2955	7.1974

Table 5.15 BW Comparison

From table 5.15, it is clear that CFMON has better bandwidth than the DBMON. Further, to obtain the throughput (TP) the author has transmitted the data packets through CFMON and DBMON. The number of data packets on each probabilistic value is shown in table 5.16.

Load Factor (p)	Total Number of Data packets
0.1	50
0.2	100
0.3	150
0.4	200
0.5	250
0.6	300
0.7	350
0.8	400
0.9	450
1.0	500

Table 5.16 Load on given Load Factors

Therefore, the throughput of CFMON and DBMON is shown in figure 5.22.

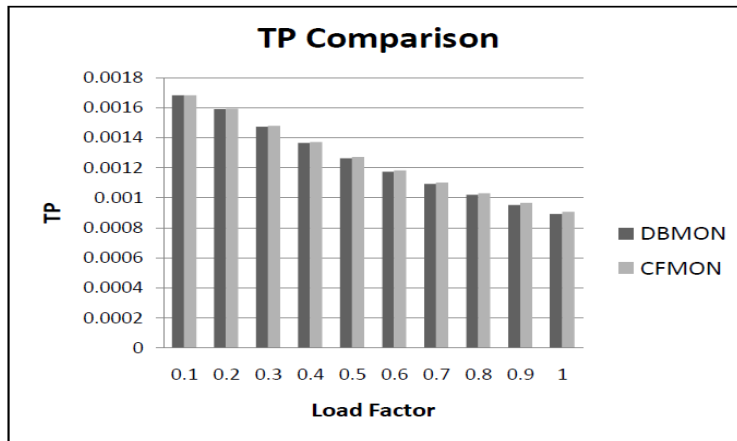


Figure 5.22 TP Comparison when N=16

From figure 5.22, it is clear that CFMON has better throughput than DBMON on each load factor (p).

5.5 Conclusion and Future Scope of the Work

Throughout the chapter, the author has discussed about the crosstalk because it is the critical issue in MINs. The whole performance of the network is also tarnished because of crosstalk. Earlier proposed approaches provide solutions against the crosstalk however, these solutions are not so effective. In this chapter the author has presented, ASA which is applicable on omega and clos networks.

ASA is effective only for the networks of small sizes. To remove the limitation of ASA the author proposed DBMON with its scheduling algorithm. The proposed network is a generalized solution against the crosstalk problem. It gives better performance than the CFMON in terms of maximum NOP and maximum transmission time. However, in terms of BW and TP, CFMON is more better than the DBMON. In future, the author wants to design the scheduling algorithm for the original omega network in order to obtain the crosstalk free routes in minimum number of passes and in minimum transmission time. The authos also wants to design a new network that has better bandwidth and throughput with minimum crosstalk.

CHAPTER 6

CONCLUSION AND FUTURE WORK

Increasing efficiency of a MIN is a critical challenge. Various performance factors are equally responsible for measuring the efficiency of a MIN. The research challenges which become the factors of low performance should be improved to obtain a good MIN. This thesis has paid its full attention to problem of switch fault and gave a little contribution to the problem of cross-talk. Both these problems decrease the efficiency of the MIN.

Further, in faulty switch problem, the network should provide the alternate path to the data packets. Various aspects have their significant role in designing a highly fault tolerant MIN e.g. cost, bandwidth, and probability of acceptance, throughput, processor utilization, and processing power. Previously proposed MINs do not possess all the aspects perfectly. In chapter 3, the author has proposed the AIAMIN, AIASEN, and IASEN-2. The AIAMIN is compared with the MALN. AIASEN and IASEN-2 are compared with IASEN. In chapter 3, it is shown that the proposed MINs have better fault tolerability than the previously proposed MINs.

In chapter 4, the author has proposed the IAON and IASEN-3. In this chapter, the author has compared the performances of IAON with MALN and IASEN-2 in single switch faulty and non-faulty conditions. It gives better performance in both conditions than the MALN and IASEN-2. Further, the performance of IASEN-3 is compared with the IAON and IASEN-2 in single switch faulty and non-faulty conditions. It is found that IASEN-3 is better than the IAON and IASEN-2 in both conditions. The chapter 4 also shows the performance of IAON in non-faulty, single switch faulty, and double switch faulty conditions. All the presented results of chapter 4 show that if the number of faulty switching elements will increase than the performance of the network will decrease.

In chapter 5, the author has addressed the crosstalk problem. It degrades the performance of a network. To reduce the crosstalk a lot of algorithms were proposed. The author has proposed address selection algorithm which is applicable on omega and clos network. The problem with address selection algorithm is that it does not perform well for the network of large sizes. Therefore, the author has proposed the destination based modified omega network and its scheduling algorithm called the destination based scheduling algorithm. Results of chapter 5 show that DBMON is better than the CFMON in terms of maximum number of passes and minimum transmission time.

In this way, the author has given his solutions in order to reduce the both problems and published his research work in this regard.

In future, the author wants to work on the following issues:

- Applying new connection pattern on the proposed MINs to handle the critical faulty switch problem.
- Designing a new scheduling algorithm for the original omega network to obtain the crosstalk free routes in the minimum number of passes and minimum transmission time.
- Designing a network that can handle both the problems, i.e. crosstalk and faulty situations simultaneously.
- Try to focus on the other issues of MINs like load balancing, routing in MINs etc.

References

1. J. Duato, S. Yalamanchili, and L. M. Ni, Interconnection networks: An engineering approach, Morgan Kaufmann, 2003.
2. K. Hwang, Advanced Computer Architecture: Parallelism, Scalability, Programmability, Tata McGraw-Hill, Noida, India, 2000.
3. W. Dally and B. Towles, Principles and Practices of Interconnection Networks, Morgan Kaufmann, San Francisco, Calif, USA, 2004.
4. H.J. Siegel, Interconnection network for large scale parallel processing: Theory and case studies, McGraw Hill, ISBN 0-07-057561-4, 1990.
5. J. Sengupta, Interconnection Networks For Parallel Processing, Deep & Deep Publications, 2005.
6. Nitin, A new approach to stable matching and networks on chip problem, Ph.D. Thesis, Computer Science & Engineering and Information Technology Department, Jaypee University of Information Technology, Solan, Himachal Pradesh, 2008.
7. V. P. Bhardwaj and Nitin, Message broadcasting via a new fault-tolerant irregular advanced omega network in faulty and non-faulty network environments, Journal of Computer and Electrical Engineering, Hindawi Publishing Corporation, vol. 2013, pp. 1-17, 2013.
8. V. P. Bhardwaj and Nitin, A new fault tolerant routing algorithm for advance irregular alpha multistage interconnection network, Advances in Intelligent and Soft Computing, Springer-Verlag, ISSN: 1867-5662, pp. 979-987, 2012.

9. V. P. Bhardwaj and Nitin, On performance analysis of IASSEN-3 in faulty and non-faulty network conditions, AASRI Conference on Intelligent Systems and Control, Elsevier, pp. 104-109, 2013.
- 10.V. P. Bhardwaj and Nitin, A new fault tolerant routing algorithm for IASSEN-2, Proceedings of the 2nd IEEE International Conference on Advances in Computing and Communications, pp. 199-202, 2012.
- 11.V. P. Bhardwaj and Nitin, A new fault tolerant routing algorithm for advance irregular augmented shuffle exchange network, Proceedings of the 14th IEEE International Conference on Computer Modeling and Simulation, pp. 505-509, 2012.
- 12.V. P. Bhardwaj and Nitin, On minimization of crosstalk conflicts in destination based modified omega network, Journal of Information Processing Systems, vol. 9, no. 2, pp. 301-314, 2013.
- 13.V. P. Bhardwaj, Nitin, and VipinTyagi, An algorithmic approach to minimize the conflicts in an optical multi-stage interconnection network, Communications in Computer and Information Science Series, Springer-Verlag, vol. 191, ISSN: 1865-0929, pp. 568-576, 2011.
- 14.V. P. Bhardwaj and Nitin, Applying address selection algorithm on nonblocking optical multi-stage interconnection network, Proceedings of the IEEE World Congress on Information and Communication Technologies, pp. 694-698, 2011.
- 15.S. Kaur, Performance of optical multi-stage interconnection networks, Masters Thesis, Computer Science & Engineering Department, Thapar University, Patiala, Punjab, 2009.

- 16.R. Mahajan and R. Vig, Performance and reliability analysis of new fault-tolerant advance omega network, WSEAS Transactions on Computers, vol. 7, no. 8, pp. 1280–1290, 2008.
- 17.S. Bataineh and G. E. Qanzu'A, Reliable Omega Interconnected Network for Large-Scale Multiprocessor Systems, The Computer Journal, vol. 46, no.5, 2003.
- 18.D. C. Vasiliadis, G. E. Rizos, S. V. Margariti, and L. E. Tsiantis, Comparative study of blocking mechanisms for packet switched omega networks, in proceedings of the 6th WSEAS international conference on electronics, hardware, wireless and optical communications, pp. 18–22, Corfu Island, Greece, 2007.
- 19.M. Abdullah, M. Othman and R. Johari, An efficient approach for message routing in optical omega network, International Journal of The Computer, the Internet and Management, vol. 14, no. 1, pp.50- 60, 2006.
- 20.R. Aggarwal, L. Kaur and H.Aggarwal, Design and reliability analysis of a new fault tolerant multi-stage interconnection network, ICGST-CNIR Journal, vol. 8, issue 2, pp. 17-23, 2009.
- 21.A. Gupta and P. K. Bansal, Fault tolerant irregular modified alpha network and evaluation of performance parameters, International Journal of Computer Applications, vol. 4, no. 1, pp.9–13, 2010.
- 22.M. Ghai, V. Chopra, and K. K. Cheema, Performance analysis of fault-tolerant irregular baseline multistage interconnection network, International Journal on Computer Science and Engineering, vol. 2, no. 9, pp. 3079–3084, 2010.

- 23.M. Abdullah, M. Othman, and R. Johari, An efficient approach for message routing in optical omega network. *International Journal of the Computer, the Internet and Management* vol. 14, no. 1, pp.50–60, 2006.
- 24.F. Abed and M. Othman, Efficient window method in optical multistage interconnection networks, *Proceedings of the 2007 IEEE International Conference on Telecommunication and Malaysia International Conference on Communications, Penang, Malaysia*, pp. 181-185, 2007.
- 25.F. Abed and M. Othman, Fast method to find conflicts in optical multistage interconnection networks, *International Journal of The Computer, The Internet and Management*, vo. 16, no.1, pp.18-25, 2008.
- 26.M.A. A. Shabi and M. Othman, A new algorithm for routing and cheduling in optical omega network, *International Journal of The Computer, the Internet and Management*, vol. 16, no. 1, pp.26-31, 2008.
- 27.S. Kaur, Anantdeep and D. Aggarwal, Effect of Crosstalk on Permutation in Optical Multistage Interconnection Networks, *Journal of Computing*, vol. 2, issue 4, pp. 100-104, 2010.
- 28.S. Kaur and N. Shukla, Reducing Crosstalk in an optical multistage interconnection network, *International Journal of Computer Applications*, vol. 46, no. 14, pp.8-13, 2012.
- 29.T.D. Shahida, M. Othman and M.K. Abdullah, Fast zerox algorithm for routing in optical multistage interconnection networks, *IIUM Engineering Journal*, vol. 11, no. 1, pp.28-39, 2010.
- 30.C. C. Fan and J. Bruck, Tolerating multiple faults in multistage interconnection networks with minimal extra stages, *IEEE Transactions on Computers*, vol. 49, no. 9, pp. 998–1004, 2000.

31. Nitin, Component Level Reliability analysis of fault tolerant hybrid MINs, WSEAS Transactions on Computers, vol. 5, no. 9, pp. 1851–1859, 2006.
32. Nitin, V.K. Sehgal and P.K. Bansal, On MTTF analysis of a fault tolerant hybrid MINs, WSEAS Transactions on Computer Research, vol. 2, no. 2, pp. 130–138, 2007.
33. Nitin and A. Subramanian, Efficient algorithms and methods to solve dynamic MINs stability problem using stable matching with complete ties, Journal of Discrete Algorithms, vol. 6, no. 3, pp. 353–380, 2008.
34. S. Sharma, P.K. Bansal and K.S. Kahlon, On a class of multistage interconnection network in parallel processing, International Journal of Computer Science and Network Security, vol. 8, no. 5, 2008.
35. R. Rastogi, Nitin and D.S. Chauhan, 3–disjoint paths fault–tolerant omega multistage interconnection network with reachable sets and coloring scheme, Proceedings of the 13th IEEE International conference on Computer Modeling and Simulation, UK, pp. 551-556, 2011.
36. R.R. Aggarwal, L. Kaur, Fault tolerance and permutation analysis of ASEN and its variant, International Journal of Computer Science and Information Technologies, vol. 1, issue 1, pp. 24-32, 2010.
37. Nitin, R. Vaish, and U. Shrivastava, 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, vol. 59, no. 3, pp. 1252–1288, 2010.

38. Nitin and D. S. Chauhan, Stochastic communication for application specific Networks-on-Chip, *Journal of Supercomputing*, vol. 59, no. 2, pp. 779–810, 2010.
39. Nitin and D. S. Chauhan, Comparative analysis of traffic patterns on k-ary n-tree using adaptive algorithms based on Burton normal form, *Journal of Supercomputing*, vol. 59, no. 2, pp. 569–588, 2010.
40. Nitin, S. Garhwal, and N. Srivastava, Designing a fault-tolerant fully-chained combining switches multi-stage interconnection network with disjoint paths, *Journal of Supercomputing*, vol. 55, no. 3, pp. 400–431, 2011.
41. M.A. Wani and H.R. Arabnia, Parallel edge-region-based segmentation algorithm targeted at reconfigurable multi-ring network, *The Journal of Supercomputing*, vol. 25, no. 1, pp. 43-63, 2003.
42. R. Bashirov and T. Karanfiller, On path dependent loss and switch crosstalk reduction in optical networks, *Information Sciences Journal*, Elsevier, vol. 180, issue 6, pp. 1040-1050, 2010.
43. A. Gupta and P. K. Bansal, Proposed fault tolerant new irregular augmented shuffle network, *Malaysian Journal of Computer Science*, vol. 24, no.1, pp. 47-53, 2011.
44. C. W. Chen and C. P. Chung, Designing a disjoint paths interconnection network with fault tolerance and collision solving, *Journal of Supercomputing*, vol. 34, no. 1, pp. 63–80, 2005.
45. C. W. Chen, Design schemes of dynamic rerouting networks with destination tag routing for tolerating faults and preventing collisions, *Journal of Supercomputing*, vol. 38, no. 3, pp. 307–326, 2006.

- 46.J. Garofalakis and E. Stergiou, An approximate analytical performance model for multistage interconnection networks with backpressure blocking mechanism, *Journal of Communications*, vol. 5, no. 3, pp. 247–261, 2010.
- 47.J. A. Sadi, K. Day and M. O. Khaoua, A fault tolerant routing algorithm for 3-D torus interconnection networks, *The International Arab Journal of Information Technology*, vol. 1, no. 0, 2003.
- 48.K. K. Cheema and R. Aggarwal, Design scheme and performance evaluation of a new fault tolerant multistage interconnection network, *International Journal of Computer Science and Network Security*, vol. 9, no. 9, 2009.
- 49.R.K. Dash, N.K. Barpanda and C.R. Tripathy, Prediction of reliability of multistage interconnection networks by multi decomposition method, *International Journal of Information Technology and Knowledge Management*, vol.1, no. 2, pp. 439-448, 2008.
- 50.S.M. Bataineh and B.Y. Allosl, Fault tolerant multistage interconnection network, *Telecommunication Systems*, 17:4, pp. 455-472, 2001.
- 51.Y. Mun, Performance Analysis of banyan type multistage interconnection network under non uniform traffic pattern, *The Journal of Supercomputing*, vol. 33, issue 1-2, pp. 33-52, 2005.
- 52.D. W. Qing and Y. E. Yu, On 1-rate and 2-rate multicast 3-stage clos networks, *Applied Mathematics-A Journal of Chinese Universities*, vol. 24, no. 2, pp. 151-156, 2009.
- 53.K. Kaur, P. Kaur, and H. Sadawarti, Performance analysis of new irregular multistage interconnection network, *International Journal of*

Advanced Engineering Sciences and Technologies, vol. 9, pp. 82–86, 2011.

- 54.D. Nitin, Component level reliability analysis of fault-tolerant hybrid MINs, WSEAS Transactions on Computers, vol. 5, no. 9, pp. 1851–1859, 2006.
- 55.B. V. S. Kumar, M. V. Rao, and M. A. R. Prasad, Adaptive fault tolerant routing in interconnection networks: a review, International Journal of Advanced Networking and Applications, vol. 02, no. 06, pp. 933–940, 2011.
- 56.Nitin, On asymptotic analysis of packet and wormhole switched routing algorithm for application-specific Networks on- Chip, Journal of Computer and Electrical Engineering, vol. 2012, Article ID216406, 27 pages, 2012.
- 57.Nitin and D. S. Chauhan, A new fault tolerant routing algorithm for MALN-2, in Eco-friendly Computing and Communication Systems, Communications in Computer and Information Science, pp. 247–254, Springer, 2012.
- 58.Nitin and D. S. Chauhan, A new fault-tolerant routing algorithm for IMABN-2, in Proceedings of the 2nd IEEE International Conference on Advances in Computing and Communications, pp. 215–218, Kerala, India, 2012.
- 59.S. Murali, D. Atienza, L. Benini, and G. De Micheli, A method for routing packets across multiple paths in NoCs with in-order delivery and fault-tolerance gaurantees, VLSI Design, vol. 2007, Article ID 37627, 11 pages, 2007.

- 60.D. Tutsch and G. Hommel, MLMIN: A multicore processor and parallel computer network topology for multicast, *Computers & Operations Research*, Elsevier, pp. 3807-3821, 2008.
- 61.R. Rastogi, Nitin, D. S. Chauhan and M. C. Govil, On stability problems of omega and 3-disjoint paths omega multistage interconnection networks, *International Journal of Computer Science Issues*, vol. 8, issue 4, no. 2, 2011.
- 62.D. Vasiliadis, G. Rizos, and C. Vassilakis, Performance study of multilayered multistage interconnection networks under hotspot traffic conditions, *Journal of Computer Systems, Networks, and Communications*, vol. 2010, Article ID 403056, 11 pages, 2010.
- 63.D. C. Vasiliadis, G. E. Rizos, and C. Vassilakisa, Performance analysis of dual-priority multilayer multistage interconnection networks under multicast environment, *Journal of Networks*, vol. 6, no. 6, pp. 858–871, 2011.
- 64.D. C. Vasiliadis, G. E. Rizos, and C. Vassilakis, Class-based weighted fair queuing scheduling on dual-priority delta networks, *Journal of Computer Networks and Communications*, vol. 2012, Article ID 859694, 13 pages, 2012.
- 65.M. Choi and N. Park, Performance analysis of fault tolerant multistage interconnection networked parallel instruments with concurrent testing and diagnosis, *IEEE Instrumentation and Measurement Technology Conference*, pp.1482-1486, 2002.