TOWARDS ENERGY EFFICIENT AND RELIABLE 3D MANYCORE CHIP ENABLED BY MACHINE LEARNING

By

SOURAV DAS

A dissertation submitted in partial fulfillment of the requirements for the degree of DOCTOR OF PHILOSOPHY

WASHINGTON STATE UNIVERSITY
School of Electrical Engineering and Computer Science

MAY 2018

© Copyright by SOURAV DAS, 2018
All Rights Reserved
To the Faculty of Washington State University:

The members of the Committee appointed to examine the dissertation of SOURAV DAS find it satisfactory and recommend that it be accepted.

Partha Pratim Pande, Ph.D., Chair

Janardhan Rao Doppa, Ph.D., Co-chair

Deukhyoun Heo, Ph.D.

Krishnendu Chakrabarty, Ph.D.
ACKNOWLEDGMENT

As I am on the brink of completing my PhD dissertation, I am grateful to my mentor and PhD supervisor Prof. Partha Pande, and thank him from my heart. His enthusiasm on scientific research has been and will be a lifelong inspiration for me now and in future.

I thank Prof. Janardhan Rao Doppa for co-advising me on Ph.D. research and guiding through his technical expertise. I greatly appreciate Prof. Deukhyoun Heo and Prof. Krishnendu Chakrabarty for sharing their valuable expertise and proving feedbacks. Their insightful comments were always helpful towards advancements.

I am indebted to all the co-authors of my publications and the funding agencies- National Science Foundation ad DARPA- to help me complete this work. Lastly, thanks to my fellow lab-mates for all the fun and amazing time we have had together in the Low Power and Robust Nano-System Design Lab for the last four years.
TOWARDS ENERGY EFFICIENT AND RELIABLE 3D MANYCORE CHIP
ENABLED BY MACHINE LEARNING

Abstract

by Sourav Das, Ph.D.
Washington State University
May 2018

Chair: Partha Pratim Pande
Co-chair: Janardhan Rao Doppa

As the demand for high performance and energy efficient computation has increased significantly, manycore chip architectures have emerged as a mainstream solution paradigm. A three-dimensional Network-on-Chip (3D NoC) that takes the advantages of amalgamation of two revolutionary technologies namely the NoC and 3D integration, improves the performance of manycore chip significantly. Existing 3D NoC architectures predominantly follow straightforward extension of regular 2D NoCs and suffer from multi-hop communications. In this context, we propose the design of 3D small-world NoC (3D SWNoC) architecture to overcome the challenges of mesh-based architectures and improve the performance of the chip.

In addition, the performance of 3D SWNoC mainly depends on the placement of cores and links. This is an instance of combinatorial optimization problem, which is computationally intractable and needs intelligent exploration of design space to reach physically plausible and near-optimal designs. We adapt a machine learning-based approach to overcome these computational challenges
and design an efficient and robust NoC architecture while ensuring significant reduction in convergence time.

The anticipated performance gain of 3D NoCs degrades in the presence of TSV failures due to fabrication limitations and workload induced stress. We analyze the reliability concerns associated with 3D ICs. We propose several mitigation techniques to counteract TSV failures, which includes VFI-based power management methodology, spare TSV allocation technique, and adaptive routing strategy. We carry out extensive experiments to characterize their performance to improve both reliability and lifetime of 3D NoCs.

Recently, monolithic 3D (M3D) integration has been proposed as an alternative to TSV-based 3D integration for designing ultra-low-power and high-performance circuits and systems. The smaller dimensions of monolithic inter-tier vias (MIVs) offer high density integration, flexibility of partitioning logic blocks across multiple tiers resulting in significant reduction of the total wire-length. In this work, we explore the design space of M3D-enabled small-world NoC architectures and present a comparative performance evaluation with TSV-based counterparts.

Finally, we summarize our contributions and outline some promising directions for future work based on the findings of this work. Future work includes incorporating machine learning approaches for on-chip security analysis and development of online mitigation techniques against external attacks.
TABLE OF CONTENTS

ACKNOWLEDGMENT ........................................................................................................ iii
ABSTRACT ......................................................................................................................... iv
LIST OF TABLES ............................................................................................................... x
LIST OF FIGURES .............................................................................................................. xi

1 INTRODUCTION ............................................................................................................. 1
  1.1 3D Network-On-Chip (3D NoC) for Manycore Systems .............................................. 2
  1.2 Small-world NoC and Design Optimization with Machine Learning ....................... 2
  1.3 Reliable 3D Manycore Chip ..................................................................................... 4
  1.4 Monolithic 3D (M3D)-based Manycore Chip Architecture ....................................... 5
  1.5 Organization of the Dissertation ............................................................................. 7

2 RELATED WORK .......................................................................................................... 8
  2.1 Design of 3D NoC Architectures .............................................................................. 8
  2.2 Optimization Approaches for 3D NoC ................................................................. 9
  2.3 Reliability Analysis and TSV Modeling Under Workload-induced Stress .............. 10
  2.4 Reliability Improvement Methodologies ............................................................... 11
  2.5 Alternative Physical Design: Monolithic 3D (M3D) Integration ............................... 14
3 SMALL-WORLD NETWORK-ENABLED ENERGY EFFICIENT 3D NOC DESIGN VIA TROUGH-SILICON-VIA INTEGRATION ................................................................. 16

3.1 Overview of 3D On-Chip Networks ......................................................... 16

3.2 Problem Description of 3D NoC Design ................................................... 17

3.3 Optimization of 3D SWNoC via Simulated Annealing ............................... 22

3.4 Optimization of 3D SWNoC Via Machine Learning ................................... 25

3.5 Irregular 3D NoC Topologies under Consideration ...................................... 34

3.6 Routing Algorithm ..................................................................................... 34

3.7 Experimental Results and Analysis .............................................................. 35

3.8 Summary of Small-world based 3D NoC Design ........................................ 53

4 COUNTERACTING ELECTROMIGRATION OF TSV-BASED 3D NOC VIA SPARE- VERTICAL LINK (SVL) ALLOCATION .......................................................... 54

4.1 Overview of Electromigration Challenge for TSV-enabled 3D System ............ 54

4.2 Problem Formulation for Spare VL Allocation Problem ............................ 59

4.3 Performance Evaluation of 3D NoCs with sVL Allocation .......................... 63

4.4 Summary of Spare Vertical Link Allocation for 3D NoC .............................. 75

5 JOINT EFFECTS OF ELECTROMIGRATION AND CROSSTALK NOISE ON TSV- ENABLED 3D NOCS ................................................................................ 76

5.1 Overview of Electromigration and Crosstalk Noise for 3D NoC .................... 76

5.2 TSV Reliability From 3D NoC Perspective .................................................. 77
5.3 Problem Formulation ................................................................. 80
5.4 Spare TSV Allocation for Reliability Improvement ......................... 82
5.5 Experimental Results and Analysis ............................................... 83
5.6 Summary of Reliability Analysis for TSV-enabled 3D NoC .................... 91

6 PERFORMANCE-RELIABILITY TRADE-OFFS FOR 3D TSV-ENABLED MANYCORE CHIP ....................................................................................................................... 93

6.1 Overview of Reliability Analysis for TSV-enabled 3D NoC .................. 93
6.2 TSV Reliability ............................................................................ 96
6.3 MTTF Improvement ...................................................................... 100
6.4 Problem Setup ............................................................................ 100
6.5 Reliability Improvement Methodologies .......................................... 103
6.6 Experimental Results and Analysis ................................................ 111
6.7 Summary of TSV-enabled 3D NoC Reliability Analysis ....................... 127

7 MONOLITHIC 3D INTEGRATION FOR 3D NOC DESIGN ..................... 128

7.1 Motivation for Monolithic 3D Integration for NoC Design .................. 128
7.2 Design Space of Monolithic 3D Integration-enabled NoC .................... 129
7.3 Optimization of M3D SWNoC Design ........................................... 135
7.4 Results and Analysis ................................................................... 138
7.5 Summary of M3D-enabled 3D NoC Design ..................................... 154

8 SUMMARY OF CONTRIBUTIONS AND FUTURE RESEARCH DIRECTIONS ...... 155
8.1 Summary of Contributions ................................................................. 155

8.2 Future Work Directions ................................................................. 157

REFERENCES ....................................................................................... 166
LIST OF TABLES

Table 3.1: Comparison of Average hop count and communication cost of 3D NoC Architectures ...................................................................................................................................................... 31
Table 3.2: Comparison of Average hop count and communication cost of 3D NoC Architectures ...................................................................................................................................................... 45
Table 5.1: Worst case and effective crosstalk capacitances for TSV bundles with different configurations ............................................................................................................................................. 84
Table 5.2: Effective crosstalk capacitance for different types of TSVs in a 3x3 grid pattern-based placement .................................................................................................................................................. 84
Table 6.1: VFI configurations with number of cores (and routers) and static V/F levels for VFI-cluster .................................................................................................................................................. 113
Table 6.2: Summary of reliability improvement factors for a 3D SWNoC ............................................................................................................................................................................................. 125
Table 6.3: Summary of implementation overheads for reliability improvement methodologies ............................................................................................................................................. 125
Table 7.1: Average hop count and best achieved communication cost for the optimized M3D SWNoC .................................................................................................................................................... 142
Table 7.2: The best achieved average hop count and communication path length for the TSV- and M3D-enabled optimized SWNoC .................................................................................................................. 147
Figure 3.1 Illustration of conventional TSV-based 3D integration. The horizontal dies integrate CPU, GPU, Memory block. MEMS, Analog and RF-circuitries block and so on. The heatsink is located at the bottom for removing generated heat from the chip. The vertical links are designed with TSVs which play the most important role for high-speed inter-die communication. On the side, the cross section of a 3D IC and TSV (Figure is not drawn to scale). ........................................ 16

Figure 3.2 Different kinds of popularly used 3D NoC architectures with TSVs acting as the vertical links. From left, three NoC architectures are 3D MESH, Concentrated or Ciliated 3D MESH, and 3D MESH-BUS hybrid NoC. The planar NoC configuration is same for all while the vertical link design vary depending on the NoC configuration. ........................................................................ 18

Figure 3.3 Illustration of small-world (SW) network. (Left) regular network, (middle) small-world (right) complete random network. The parameter p refers to amount of randomness in the network. For regular network p = 0, while p = 1 for complete random network. In a SW network, there are only few long-range links which induce finite randomness (0 < p < 1) [99]. ......................... 19

Figure 3.4 Illustration of small-world network-based 3D NoC (3D SWNoC). Planar layers are designed with conventional wires while the vertical links are designed with through-silicon-vias (TSVs). In as 3D SWNoC, TSVs act as long-range shortcuts necessary for inducing small-world effects. .............................................................................................................................. 21

Figure 3.5 Simulated annealing-based 3D SWNoC design and optimization steps for achieving the best parametric values for network connectivity parameter, α. ......................................................... 24
Figure 3.6 Overview of STAGE-based optimization algorithm overview. Base- and Meta-search cooperate with each other to explore better solution in a short time. ................................. 26

Figure 3.7 Instantiation of STAGE-based optimization algorithm for 3D SWNoC link placement. Two searches viz. the base search, A, and meta search, M, iterate in between them by seeking guidance from the evaluation function, E, to reach to the optimized NoC architecture quickly. . 28

Figure 3.8 Variation of normalized EDP with respect to the small-world connectivity parameter $\alpha$. The EDP is normalized with respect to the 3D MESH for each respective benchmark. .................. 36

Figure 3.9 Average normalized EDP of 9 benchmarks of 3D SWNoC with variation in connectivity parameter, $\alpha$ (normalized to 3D MESH). ................................................................. 37

Figure 3.10 Percentage of total traffic carried out by each layer (Die 1 to 4) before and after task remapping. ........................................................................................................... 38

Figure 3.11 Best objective value $O$ and error rate of the evaluation function $E$ for the STAGE algorithm over iterations. ................................................................................................. 39

Figure 3.12 Performance comparison between the machine-learning-based optimization algorithm (STAGE) and the Simulated Annealing (SA) algorithm. .......................................................... 40

Figure 3.13 Effect of optimization algorithm on weighted communication ($w_{ij} = \text{sum of all } (f_{ij} \times h_{ij})$) features. .......................................................................................................................... 42

Figure 3.14 Normalized network latency per message of 3D SWNoC before and after optimization. .................................................................................................................. 43

Figure 3.15 Normalized energy consumption per message of 3D SWNoC before and after optimization. .................................................................................................................. 44

Figure 3.16 Normalized EDP per message of 3D SWNoC before and after optimization........ 44
Figure 3.17 Normalized network latency of different 3D NoCs (3D MESH, *mrrm*, *rrrr*, and 3D SWNoC). All the values are normalized with respect to that of 3D MESH for respective benchmark.

Figure 3.18 Normalized energy consumption per message of different 3D NoCs (3D MESH, *mrrm*, *rrrr*, and 3D SWNoC). All the values are normalized with respect to that of 3D MESH for respective benchmark.

Figure 3.19 Normalized EDP per message of different 3D NoCs (3D MESH, *mrrm*, *rrrr*, and 3D SWNoC). All the values are normalized with respect to that of 3D MESH for respective benchmark.

Figure 3.20 3D SW NoC performances normalized to fault free (fully vertical connected) 3D MESH Vs. the percentage of vertical link failure rate. (a) Average normalized network latency, (b) Average energy consumption per message, (c) Average EDP.

Figure 3.21 Network latency, energy consumption and EDP per message of the 3D SW_partial. All the values are normalized to that of fault free (fully connected) 3D MESH.

Figure 3.22 Normalized EDP profile of 3D NoCs with workload induced vertical link failure scenario for CANNEAL benchmark. The EDP is normalized with respect to fault free 3D MESH at t=0.

Figure 4.1 (left) Workload induced stress and electromigration phenomenon for the TSVs. The through current causes stress and forces the TSV material to undergo electromigration at the landing pad and creates void. (right) TSV resistance profile with time.

Figure 4.2 Effect of non-homogeneous TSV utilization for 3D SWNoC with CANNEAL benchmark. (a) TSV utilization (in percentage of cycles that TSV actively transmit signals to total simulated cycles). (b) MTTFs for individual TSVs.
Figure 4.3 (a) TSV-based vertical link placement between routers from adjacent tiers. TSVs are placed in bundle e.g. 3x3, 4x4 etc. configuration. (b) TSV model to study the electromigration effect. .......................................................................................................................... 57

Figure 4.4 (a) Spare TSV allocation explained. Spare TSVs are placed close to the functional TSVs and replaces any possible failed functional TSV whenever necessary. (b) In spare vertical link allocation, a group of spare TSVs forms the spare vertical link (spare VL) and replaces the whole bundle of TSV-enabled functional VL. In this case, we focus on spare VL allocation. .......................... 58

Figure 4.5 Lifetime determination algorithm is explained with normalized EDP profile of 3D SWNoC w/ and w/o sVL allocation. As an example, lifetime calculation for 3D SWNoC with DEDUP benchmark has been plotted. The EDP of 3D MESH-0 sVL (dotted line) corresponds to time t=0 and extended only for reference purpose. .................................................................................. 65

Figure 4.6 Lifetime of the 3D SWNoC as a function of the number of sVL for the CANNEAL benchmark: Greedy-Full vs. Exhaustive-Full. ............................................................................................................. 66

Figure 4.7 (a) Runtime for different number of sVL: Greedy-Full vs. Greedy-Restricted. (b) Runtime for different number of sVL: Exhaustive-Full vs. Exhaustive-Restricted. ................. 69

Figure 4.8 Normalized EDP profile for 3D SWNoC with different number of spare VLs allocation for the (a) CANNEAL, (b) DEDUP, (c) VIPS benchmark................................................................. 71

Figure 4.9 Effect of sVL allocation on 3D SWNoC for the CANNEAL benchmark. The improvement of lifetime of 3D SWNoC initially increases linearly and saturates beyond 8-sVL allocation. The gain is normalized w.r.t. the initial lifetime of 3D SWNoC at t=0. ................. 74

Figure 5.1 Different kinds of TSV placement configuration. Most popular approach is the grid-based placement (e.g. 3x3, 3x4, 4x5, 3x5 and so on). ................................................................. 77
Figure 5.2 Placement of TSVs in a grid. (a) The worst-case crosstalk capacitance for the center TSV (TSV 5). (b) The aggregation of worst case $C_c$ values for each TSV. ............................. 78

Figure 5.3 Equivalent circuit model of TSV to determine the effects of workload-induced stress and crosstalk................................................................................................................. 82

Figure 5.4 Variation of crosstalk capacitances for different TSV grid-based placement configurations ........................................................................................................................................... 85

Figure 5.5 Variation of MTTF for three types of TSVs in 3x3 grid configuration for (a) CANNEAL, (b) DEDUP, and (c) VIPS benchmark........................................................................................................................................................................... 87

Figure 5.6 Effects of only workload-induced stress (EM-only) and joint effects of EM and crosstalk capacitances (EM + $C_c$) on TSV MTTF distribution for CANNEAL benchmark........ 88

Figure 5.7 Improvement of lifetime of 3D SWNoC with the spare TSV allocation algorithm by considering only workload-induced stress (sTSV_EM) and the joint effects (sTSV_EM_$C_c$).... 89

Figure 5.8 Lifetime of TSV based 3D SWNoC with different number of sTSVs considering the joint effects of workload–induced stress and crosstalk............................................................................................................................. 91

Figure 6.1 Illustration of the reliability-performance trade-off analysis for TSV-enabled 3D NoCs. .............................................................................................................................................................. 95

Figure 6.2 Workload variation and its effects on the vertical links (VLs) of a 64-core 3D SWNoC for the DEDUP benchmark. (a) Time average VL utilization metric ($v_{\text{avg}}$) for all the VLs numbered from 1 to 48. (b) The MTTF distribution for the same system. All the MTTF values are normalized with respect to the lowest MTTF value in the system. The maximum MTTF can be as high as 6.2 times of the lowest MTTF........................................................................................................................................... 97
Figure 6.3 Illustration of DVFI implementation methodology for 3D SWNoC, and the establishment of performance-reliability trade-off. Incorporating DVFI strategy improves both EDP profile and the overall reliability (lifetime) by improving the VL MTTF values. ............ 107

Figure 6.4 Illustration of ALASH based adaptive routing strategy for 3D SWNoC, and establishment of performance-reliability trade-off. In adaptive routing, the main goal is to increase the MTTF values of the highly utilized VLs incurring small EDP penalty. The parameter, $\sigma$, denotes the standard deviation in the MTTF values at time $t$. ................................................................. 109

Figure 6.5 (a) Spare TSV allocation with 50% additional TSV or for fTSV to sTSV ratio 4:2 from [Kang et al. 2010]. (b) Spare TSV allocation for the vertical links of 3D NoCs with 4:2 approach .......................................................................................................................... 111

Figure 6.6 Normalized EDP profiles of 3D SWNoC with adaptive routing (Adaptive) and VFI based power management (DVFI). The EDP profile is normalized with respect to a 3D MESH. ........................................................................................................................................ 114

Figure 6.7 Temperature profile of 3D SWNoC in degree Celsius ($^\circ$C) with adaptive routing (Adaptive) and VFI-based power management (DVFI). (a) Average temperature, and (b) the maximum temperature for different PARSEC and SPLASH-2 benchmarks. .................... 116

Figure 6.8 The MTTF distribution profile of TSV enabled vertical links (VLS) for 3D SWNoC with VFI based power management and adaptive routing strategy for four benchmarks viz. (a) CANNEAL, (b) FFT, (c) DEDUP, and (d) VIPS. The horizontal axis indicates the normalized MTTF values while vertical axis refers to percentage of occurrence for a particular MTTF value. The MTTFs are normalized with respect to the lowest MTTF value of the system for each respective case. ......................................................................................................................................................................................... 119
Figure 6.9 Normalized lifetime of 3D SWNoC with VFI based power management, adaptive routing strategy, and with 4:2 spare TSV allocation. All the lifetime values are normalized with respect to the lifetime of the baseline 3D SWNoC. ................................................................. 123

Figure 7.1 (a) Illustration of Monolithic 3D (M3D) and TSV-based 3D structure. While two thick layers i.e. bonding layer and die substrate are present between the tiers for TSV-based integration, however, for M3D-enabled architecture, only thin inter-layer dielectric (ILD) layer is present. (b) Comparison of physical dimensions of TSV and MIV vias. The figure is redrawn from [84]. . 128

Figure 7.2 Illustration of a small-world (SW) network-enabled 64-node 3D NoC architecture with M3D- and TSV- integrations. For simplicity, only the router tiles and the network connectivities are shown (associated cores connected to routers are not shown). For M3D-enabled architecture, routers extended over multiple-tiers are colored with (light) orange while the single-tier routers are shown in blue. ........................................................................................................... 130

Figure 7.3 Illustration of the design space and optimization complexity comparison between TSV- and M3D-integration-based small-world (SW) network-enabled 64-node 3D NoC architectures. .................................................................................................................................................. 134

Figure 7.4 Instantiation of STAGE-based 3D SWNoC architecture optimization approach enabled with monolithic integration. The base search, $A$, optimizes the objective function, $O$, while meta search, $M$, guides the base search to initial solution space that has promise to lead better quality solution. The evaluation function, $E$, is a regression learner that learns the design space of M3D-enabled NoC........................................................................................................................................... 135

Figure 7.5 (a) Best-achieved communication cost for the M3D SWNoC during the link and router placement optimization with STAGE- and simulated annealing (SA)- algorithms. (b) Convergence
time improvement for STAGE-based link- and router-placement optimization over SA-based approach for M3D- and TSV-enabled SWNoC architecture. ................................................. 140

Figure 7.6 (a) Normalized network latency, (b) EDP per message (w.r.t. M3D MESH) of random and optimized M3D SWNoC .................................................................................. 141

Figure 7.7 (a) Normalized runtime of STAGE and SA algorithm for NoC link and router placement optimization, (b) Normalized runtime of SA for different NoC size with extrapolation for larger system. ........................................................................................................ 144

Figure 7.8 (a) Normalized network latency, (b) EDP per message (w.r.t. M3D MESH) of M3D-enabled MESH, mrrm, rrrr, and SWNoC architectures. .................................................. 145

Figure 7.9 Normalized (a) energy per message, (b) EDP per message of an optimized 3D SWNoC with TSV- and M3D-enabled designs (normalized w.r.t. TSV-based design). ...................... 146

Figure 7.10 (a) Normalized EDP value of different benchmarks, (b) Average of the normalized EDP value of all benchmark (w.r.t. the highest EDP value) of M3D- SWNoC with connectivity parameter variation ......................................................................................... 148

Figure 7.11 Normalized (a) energy, (b) EDP per message of M3D- SWNoC with single and multi-tier routers. ............................................................................................................... 149

Figure 7.12 Normalized (a) EDP for different benchmarks, (b) average EDP per message of M3D-SWNoC different percentage of links reduction ........................................................................ 151

Figure 7.13 Normalized average energy and EDP per message of M3D-SWNoC with different percentage of additional routers incorporated in the NoC architecture. ......................... 153

Figure 8.1 Explanations of three different kinds of NoC attack scenario depending on the NoC domain and architectural expertise of the attacker. For uniform random attack, the hacker only needs the access to user end and manycore chip architecture knowledge. For critical region attack,
the knowledge regarding the network configuration is required, while the router architecture and TSV placement configurations are necessary for critical VL attack. ........................................ 159

Figure 8.2 Different reliability concerns for Monolithic integration and reliability improvement-aware design optimization framework................................................................. 164
Dedication

To my parents, Ardhendu Das and Sushama Das, who helped me in every little step on my journey to here, and to all my family members.
1 INTRODUCTION

The demand for high-performance and low power computing has grown exponentially in recent years due to emerging applications including image processing, deep learning, graph analytics, and other big data computing problems. Manycore chip architectures offer a low-power and highly scalable platform suitable for these computation and communication intensive applications. In this context, the introduction of network-on-chip (NoC) architecture to separate the communication fabric from the computational units has revolutionized the multicore chip design [1] [2] [3] [4]. Traditionally, researchers focused on efficient 2D NoC architecture development and such efforts improved the performance significantly [5] [6]. Examples of such 2D NoC architectures are MESH, SPIN, TORUS, Folded-TORUS, Octagon, BFT etc. [6] [7]. However, as the system size grows, the chip footprint increases and the number of global wires also increases. The power consumption of such long wires becomes a significant portion of the total power of the chip budget. In addition, restricted floor-planning choices of the 2D integrated circuits (ICs) also impose limitations for the performance enhancements achieved from the NoC architectures [7] [8]. Hence, to improve the performance of NoC architecture, new technology is required that can integrate larger number of cores in a single chip to minimize the power consumption and improve the performance.

To overcome the limitations associated with the 2D NoC topology, researchers introduced the idea of die stacking and integration of large number of cores using the vertical direction. Three-dimensional (3D) ICs are capable of achieving better performance, functionality, and packaging density compared to the traditional planar ICs. The number of horizontal long global interconnects
reduces, which increases the speed [9] [10] [11]. The wire capacitance and number of repeaters in the interconnect also decreases significantly resulting in the reduction of interconnect power [11] [12]. The increased connectivity also helps in improving the performance of NoC.

1.1 3D Network-On-Chip (3D NoC) for Manycore Systems

With freedom in the third (vertical) dimension, NoC architectures that were previously impossible or prohibitive due to wiring constraints in planar ICs are now realizable in 3D NoC, and many 3D implementations can outperform their 2D counterparts. Examples of such 3D NoCs are 3D MESH, Ciliated-MESH, Concentrated MESH etc. [8] [7]. It is found that for a 64-core system, 3D MESH can outperform its 2D counterpart by 25% and 40% in terms of energy per message and EDP respectively [7]. However, existing 3D NoC architectures predominantly follow straightforward extensions of regular 2D NoC designs, which do not fully exploit the advantages provided by the 3D integration technology [13]. Consequently, we target to explore the design space of energy efficient and high-performance 3D manycore chip architectures.

1.2 Small-world NoC and Design Optimization with Machine Learning

The additional degree of freedom provided by the vertical interconnects of 3D ICs enables the design of more efficient irregular architectures. In this context, design of small-world network-based NoC architectures [14] is a notable example. A small-world network lies between a completely regular and random network by incorporating finite number of long-range links. It is already shown that either by inserting long-range shortcuts in a regular mesh architecture to induce small-world effects or by adopting power-law based small-world connectivity it is possible to achieve significant performance gain and lower energy dissipation compared to traditional multi-
hop mesh networks [14] [15]. In this work, we advocate that this concept of small-worldness should be adopted in 3D NoCs too. More specifically, the vertical links in 3D NoCs should enable design of long-range shortcuts necessary for a small-world network. These vertical connections give rise to close proximity among communicating nodes that would have been far apart in a solely planar system.

In this work, we propose design of 3D small-world (SW) NoC architectures [16]. Considering the perfectly aligned vertical link placement constraints of Through Silicon Vias (TSVs), we first determine the suitable design parameters of the 3D SWNoC. Next, by exploiting the vertical dimension in a 3D IC, the tasks are mapped among the cores in such a way that physically long distant and highly communicating cores are placed along the vertical dimension, and hence overall system performance can be significantly improved [17].

The design space of a 3D SWNoC is combinatorial in nature. The achievable performance of 3D SWNoC and other irregular NoC topologies depends on the placement of the links and routers, and overall how efficiently the architecture is optimized for a given set of objective function and constraints. Hence, the architecture needs to be optimized for link and core placement to ensure energy efficiency and high performance.

Machine learning techniques have seen great success in diverse application domains, particularly solving complex optimization problems [18] [19]. In this work, we leverage machine-learning techniques to intelligently explore the design space to optimize the placement of both planar and vertical communication links for high performance and energy efficiency. In addition, to lower the power consumption overall chip, we employ machine learning inspired power management
strategy. We demonstrate that machine learning inspired approaches can achieve high quality design solutions while ensuring low convergence time [20].

1.3 Reliable 3D Manycore Chip

In state-of-the-art technology, TSVs are predominantly used as the vertical communication links between planar dies due to their reduced length and higher integration density [21] [22]. However, the anticipated performance gain of any 3D NoC-enabled manycore chip will be compromised due to the potential failures of TSV based vertical links (VLs) (a VL corresponds to a bundle of TSVs) [23] [24]. TSVs in a 3D system can fail due to the formation of voids, cracks, misalignments, improper bonding, thermal hotspots, electromigration etc. [25] [26] [27]. The non-homogeneous workload-induced stress among the TSV-enabled vertical links (VLs) in 3D NoCs causes large variations in the mean-time-to-failure (MTTF) distribution. Such a failure of TSV-based VLs leads to performance degradation over time [28] [29] [30] [31]. One possible solution to enhance the lifetime of these VLs is to incorporate a suitable power management strategy.

To enhance the reliability of 3D NoC-enabled manycore chips, we propose several solutions and compare their performance. These includes spare TSV allocation, adaptive routing to homogenize the TSV utilization and to adopt the voltage-frequency island (VFI)-based power management methodology. Among all of them VFI-based power management has been identified as the most compelling solution. The VFI-based approach helps to reduce the energy consumption and hence, the workload-induced stress of the highly utilized VLs. The adopted power-management strategy relies on control decisions about the voltage/frequency (V/F) levels on VLs. We demonstrate that compared to the well-known spare allocation and adaptive routing strategies, power management is more effective in enhancing the reliability of a 3D NoC. VFI-based power management
improves the reliability of the 3D NoC by one order of magnitude compared to both adaptive routing and spare allocation while running popular SPLASH-2 and PARSEC benchmarks. The principal benefit of power management is that it is capable of reducing the operating temperature of the system, which in turn enhances the Mean-Time-To-Failure (MTTF) of the VLs and reliability of the overall 3D NoC.

1.4 Monolithic 3D (M3D)-based Manycore Chip Architecture

Monolithic 3D (M3D) integration, a breakthrough technology to achieve “More Moore and More Than Moore,” opens up the possibility of designing cores and associated network routers using multiple tiers by utilizing monolithic inter-tier vias (MIVs) and hence, reducing the effective wire length [31] [30] [32] [33]. Compared to TSV-based 3D ICs, M3D offers the “true” benefits of vertical dimension for system integration: the size of an MIV used in M3D is over 100x smaller than a TSV [34] [35] [36]. This dramatic reduction in via size and the resulting increase in density opens up numerous opportunities for design optimizations in 3D NoCs.

The International Technology Roadmap for Semiconductors (ITRS) projects TSV pitch remaining in the range of several microns, while on-chip interconnect pitch will be in the range of 100 nm or less. The keep-out-zone (KOZ) required for TSVs and limitations on the die alignment precision impose limits on the achievable device integration density by using TSV-based 3D stacking. A minimum KOZ of 3 μm is required for ICs fabricated at the 20-nm technology node [37] [38], and the die alignment precision is currently limited to 0.5 μm [39]. Therefore, the TSV pitch will not reduce appreciably in the future due to bonder alignment limitations (0.5-1 μm) and stacked silicon layer thickness (6-10 μm). While TSV pitches in the micron range may provide a sufficient number of vertical connections for stacking memory atop processors and memory-on-memory stacking,
they may not be sufficient to significantly mitigate the on-chip interconnect problems. Monolithic 3D ICs offer vertical connections with less than 50 nm diameter and hence, provide 10,000 times the area density of TSVs [30].

On the other hand, NoC is an enabling solution for integrating large numbers of embedded cores in a single die. 3D NoC architectures combine the benefits of these two paradigms (3D IC and NoC) to offer an unprecedented performance gain even beyond the Moore’s law regime. Existing 3D NoC architectures mainly incorporate simple extensions of regular 2D NoCs. However, this approach does not fully exploit the advantages provided by M3D integration. In this context, the design of small-world network-based NoC architecture is a notable example [14]. It has been shown that either by inserting long-range shortcuts in a regular mesh to induce small-world effects or by adopting power-law based small-world connectivity, we can achieve significant performance gain and lower energy dissipation compared to traditional multi-hop mesh networks [14] [20]. We advocate that this concept of small-worldness should be adopted in M3D NoCs. By exploiting the MIV-based vertical connections in M3D, the multi-hop long-range planar links can be placed along the shorter Z-dimension, and hence, overall system performance can be significantly improved.

Motivated by the promise of M3D and the potential of 3D NoCs that use MIVs, we propose a design methodology for M3D-enabled high performance and energy-efficient NoC architectures. The proposed approach explores the possibility of designing an efficient NoC enabled by true multi-tier routers. We also undertake a detailed performance evaluation between M3D- and TSV-enabled 3D NoCs. To the best of our knowledge, this is the first effort that targets design and optimization of NoCs using M3D-integration.
1.5 Organization of the Dissertation

Rest of this dissertation is organized in the following way: In Chapter 2, we discuss the prior work done in the areas of 3D NoC design, optimization, and reliable manycore chip design. Chapter 3 outlines the target problem of 3D NoC design, introduces small-network-enabled 3D NoC design steps, and proposes machine-learning-inspired algorithms to optimize NoC configurations. Chapter 4 explores the reliability concerns generated from electromigration and workload-induced stress for TSV-enabled 3D NoCs and proposes a solution approach to improve the reliability profile. In Chapter 5, we incorporate the effects of crosstalk noise to study the details of TSV modeling considering the NoC design and performance constraints. Chapter 6 explores different solution methodologies to improve the lifetime of TSV-based links including spare TSV allocation, power management and adaptive routing strategy. In Chapter 7, we discuss the monolithic 3D integration (M3D)-based 3D NoC design and optimization techniques to improve the energy efficiency of manycore chip. Lastly, Chapter 8 summarizes the salient features of this work and discusses the future directions originating from this dissertation.
2 RELATED WORK

2.1 Design of 3D NoC Architectures

3D NoCs integrated the benefits of 3D integration and NoC architectures to offer unprecedented performance gain over planar counterparts [7]. Most of the existing 3D NoC architectures utilize a conventional mesh [7] [40] [41]. However, it is well-known that mesh-based architectures suffer from high network latency and energy consumption due to its multi-hop communication links. To exploit the reduced distance along the vertical dimension of 3D IC, NoC-bus hybrid architecture was proposed in [42] that uses Dynamic Time Division Multiple Access (dTDMA) to reduce network latency. To reduce energy consumption of the system, the 3D Dimensionally Decomposed (DimDe) NoC router architecture [43] was developed. Reducing the number of input ports, an improved version of 3D NoC router architecture was developed in [44]. All of these architectures have buses in the Z-dimension; and hence, with increase in the network size, they are subject to traffic congestion and high latency under high traffic injection loads.

Despite recent advances in TSV technologies, TSVs are still subject to manufacturing defects and wearout [28] [45], so researchers have developed NoCs with partial vertical connections [46]. To compensate for the loss in performance due to TSV failure, fault tolerant router and NoC architectures with redundant vertical links were proposed [47]. However, these designs give rise to additional area and power overheads.

The Sunfloor 3D was developed for synthesizing application specific 3D NoCs [48]. The design of application-specific 3D NoC architectures was also investigated in [49] [50]. Later, more
general-purpose 3D NoC was proposed in [51] using an ILP based algorithm to insert long-range links to develop low diameter and low radix architecture. However, the reduction in energy consumption was found to be limited.

Photonic interconnects offer high bandwidth and low power for future multi-core chip design. A number of hybrid 3D/photonic NoC architectures [52] [53] [54] were designed considering these benefits. However, on-chip photonics still suffer from performance variation due to thermal issues [55] [56]. In addition, the challenges of integrating two emerging paradigms, namely 3D IC and silicon nanophotonics, are yet to be adequately addressed.

In this context, we focus on designing a robust 3D NoC architecture that combines the benefits of 3D ICs and the robustness of the SW architecture. We present a detailed design methodology by incorporating small-world network for development of energy-efficient 3D NoC architectures.

2.2 Optimization Approaches for 3D NoC

Researchers have developed several 3D NoC architectures to improve the performance of 3D ICs [57] [7]. For this purpose, traditional NoC researchers have mostly employed simulated annealing (SA) or integer linear programming (ILP) as the optimization workhorse [58] [49] [51]. One of the main drawbacks of SA-based optimization is that convergence tends to be slow and the time required to reach (near-) optimal solutions increases exponentially with system size [58]. ILP solvers internally use some form of search (e.g., Branch-and-Bound) but they do not scale well for large problem sizes. Indeed, ILP based solutions have been used in the past for placing the long-range links. The resulting NoC was shown to have very small network latency, but it suffered from relatively high-power consumption [49]. In this work, to overcome the above drawbacks, we
employ optimization algorithms based on machine learning (ML) that perform intelligent search to reduce the convergence time and improve the quality of the overall NoC design.

2.3 Reliability Analysis and TSV Modeling Under Workload-induced Stress

Several studies have addressed TSV failures in a 3D IC. Prior work focused on exploring the types of TSV faults, analytical and statistical modeling of TSV failures, developing strategies to reduce the number of fabrication and bonding defects, handling timing faults, and improving chip yield rates [59] [60]. Consequently, fault tolerant architectures were developed [47], and spare TSV (sTSV) allocation strategies were proposed to improve reliability [23] [61].

Several studies have also explored electromigration (EM) induced TSV failures and modeling of TSVs under EM [26] [62] [63]. The effects of externally applied stress on EM were analyzed for TSV materials and the corresponding bonding wires [60]. The authors in [64] explored clock tree synthesis under EM faults and evaluated their impacts on clock-delay. Similarly, the effects of EM were analyzed on power and ground lines for a 3D IC [65]. A bundle-based spare TSV (sTSV) placement method and mapping strategy between the functional TSVs (fTSVs) and sTSVs were proposed for EM faults to achieve a target mean-time-to-failure (MTTF) for the system [66].

From the NoC design perspective, it has been demonstrated that a small-world network-enabled 3D NoC (3D SWNoC) is extremely robust against VL failures [16]. The effect of workload-induced stress on the TSV-based vertical links of a 3D NoC has also been addressed [29]. In particular, due to the non-homogeneous workload, heavily used TSVs wear-out quickly and contribute to the wear-out of neighboring TSVs. Hence, the MTTF of those TSVs will decrease,
which will adversely affect the overall lifetime of the chip. It is also shown that by employing adaptive routing, the MTTF of a 3D NoC can be improved significantly [29].

In addition, crosstalk severely affects the delay, and consequently the MTTFs of the TSVs. Hence, if we only consider the workload-induced stress for analyzing the 3D NoC reliability, we would be targeting only one part of the overall problem. The combined effects of the electromigration arising out of workload-induced stress and the crosstalk capacitance need to be considered to determine the realistic lifetime of a 3D NoC. Consequently, any sTSV allocation method to improve 3D NoC reliability must consider these two factors in conjunction. Hence, in this paper, our focus is to evaluate the combined impact of electromigration and crosstalk on TSV MTTF and 3D NoC reliability.

2.4 Reliability Improvement Methodologies

To improve the reliability profile of TSV-enabled system, several solutions have been proposed among which spare TSV allocation, VFI-based power management and adaptive routing strategy are one most popular approach from the 3D NoC perspective [67] [68] [69].

2.4.1 Works Related to Spare TSV Allocation

To overcome performance penalties arising from TSV failure, researchers have investigated spare TSV (sTSV) allocation and sharing scheme for 3D ICs [23] [70]. The initial sharing algorithm was developed based on the idea of utilizing one extra TSV for each of the functional TSVs [70]. However, the reliability improvement comes at the expense of double TSV count and significant area overhead.
To avoid the 100% TSV area overhead, researchers have proposed several TSV sharing schemes (e.g., 3:2, 4:1, and so on) [71] [72]. The main idea is to share spare TSV/s among a group of functional TSVs to compensate for performance penalty due to possible TSV failure within that particular group. However, depending on the sharing scheme, a significant amount of encoder and decoder logic circuits are necessary to shift the signals and select the correct spare TSV, which introduces additional delay and power consumption. In addition, the delay for each TSV can be different depending on the location of the failed TSV, which may result in timing violation. To address the varying delay for each TSV, a group-based 6-TSV placement scheme for 4 functional TSVs was proposed to improve the reliability of a 3D DRAM [73]. With the help of a switchbox-based design for each group, correct signals were selected and transferred for functional TSVs. The main advantage was that the same amount of delay was incurred by every TSV in the box. However, this advantage comes at the expense of 50% area overhead and significant power consumption from the switch-boxes. Similarly, researchers have developed a block-based redundancy architecture and used signal-shifting techniques for fault tolerance [22] [74]. In this case, if any TSV fails, then the signal shifts towards the redundant ones. The signal shifting technique can tolerate one TSV failure. To improve fault tolerance for more than one TSV failure, a crossbar-based redundant TSV architecture was developed in [61]. However, the number of redundant TSVs increases significantly in this case. TSV resource sharing algorithms, which can be selectively applied depending on the sharing granularity and design complexity were also developed. Word-level and bit-level TSV sharing was formulated as a constrained clique-partitioning problem and efficient algorithms were designed to solve it. However, these algorithms do not scale for large-scale design problems.
All the above-mentioned TSV sharing schemes improve the performance of 3D ICs and hence, the overall reliability as well. However, the allocation of spare TSVs for 3D NoCs need to consider additional constraints arising from the physical NoC design perspective. In a 3D NoC, TSVs are placed in a bundle to enable a single vertical link (VL). Depending on the physical placements of routers and cores of 3D NoCs, these VLs maintain considerable physical distance between them. Hence, sharing TSVs among these VLs is not feasible due to the physical design and timing constraints. In addition, if one TSV fails in a VL, then the achievable performance of the whole link is affected, which in turn degrades the overall NoC performance. Hence, we focus on spare-vertical link (sVL) allocation instead of individual spare TSVs.

In this work, to analyze the reliability issues of 3D NoC, we evaluate the performance of a 3D NoC with workload-induced VL failure. We perform a comparative reliability analysis study of different 3D NoCs with VL failure. Then we describe the sVL allocation mechanism for a 3D NoC. We formulate sVL allocation as an optimization problem to minimize the performance penalty due to TSV-based VL failure. At the same time, we focus on improving the lifetime of the overall 3D NoC. We demonstrate two different algorithms viz. greedy and exhaustive search to allocate the sVLs in a 3D SWNoC to compensate for the performance penalty due to VL failure.

### 2.4.2 Other Reliability Improvement Techniques

Prior work has analyzed the effects of stress due to high current flow in the TSVs and demonstrated that such stress can increase the resistance of TSVs due to electromigration and ultimately, leads to TSV failure [75] [63] [26]. As a result, the performance of the 3D NoC degrades over time. To overcome performance degradation induced by TSV failure, a fault-tolerant 3D NoC architecture was proposed in [47]. The TSV stress due to high current can be reduced if suitable power-
management strategies are incorporated to lower the operational voltage and frequency of specific VLS. For example, dynamic voltage-frequency-scaling (DVFS) and voltage-frequency islands (VFI) have been used to lower the energy consumption of manycore chips [76] [77]. While DVFS offers more fine-grained V/F tuning over the VFI-based methodology, it is not scalable for large systems. DVFS implemented at the granularity of each core and network element introduces significant hardware overhead in terms of voltage regulators and clock synchronizers [76]. In contrast, VFI-based power management is more efficient and scalable [76] [78], and hence, incorporated in modern-day manycore chips [79] [80].

In this work, we propose to incorporate VFI-based power management in 3D NoCs to improve reliability. Note that we are not proposing any new VFI methodology; instead, our goal is to exploit existing DVFI methodologies to improve the reliability of 3D NoCs. To the best of our knowledge, this is the first attempt to incorporate VFI-based power management for handling TSV failures in 3D NoCs, and to quantify the trade-off between performance and reliability. In addition, we also propose to incorporate the adaptive routing strategy to improve the lifetime of TSV-based vertical links in a 3D manycore chip.

2.5 Alternative Physical Design: Monolithic 3D (M3D) Integration

In order to improve the performance of 3D-integrated systems, monolithic 3D integration has been studied as an alternative to TSV-based designs [33]. Several semiconductor companies (e.g., GLOBAL-FOUNDRIES and Qualcomm in collaboration with CEA-LETI) are exploring monolithic 3D (M3D) as a means for overcoming the pitch and alignment limitations of TSVs [38] [81]. In order to ensure that M3D integration is primed for use when industry is ready to adopt it, researchers have developed a low-temperature process to fabricate high-performance top transistor
layers without degrading the performance of bottom transistor layers [82] [83]. Design techniques to reduce interconnect length, critical path delay, and die area were presented in [83] [84]. An M3D IC can be partitioned at the core-, block-, gate- and transistor-level. Gate-level partitioning appears to be the most promising [85] [86] [87] and it is especially relevant to 3D NoC design. Therefore, significant research efforts are being directed towards the development of design tools for gate-level-partitioned M3D ICs [86]. In this context, we advocate the incorporation of gate-level partitioning offered by the M3D-integration to design efficient SWNoC architecture. Our target is to demonstrate how the multi-tier routers based on MIVs enhance the performance and energy efficiency of 3D NoC.
3 SMALL-WORLD NETWORK-ENABLED ENERGY EFFICIENT 3D NOC DESIGN VIA TROUGH-SILICON VIA INTEGRATION

3.1 Overview of 3D On-Chip Networks

By stacking multiple dies on top of each other, 3D integration offers advantages of scaled down system footprint, reduced interconnect length, improved form factor, lower power dissipations, and overall significant performance gain over conventional planar ICs. The lower parasitic (capacitance) values and consequent reduced RC delays compared their planar counterparts result in higher operating frequencies opening the possibility of improving performance further [21] [88] [89]. Moreover, 3D integration also enables the design of heterogeneous integration of different types components within a single-chip [90] [91]. Among different alternative choices of 3D

![Figure 3.1 Illustration of conventional TSV-based 3D integration. The horizontal dies integrate CPU, GPU, Memory block, MEMS, Analog and RF-circuitries block and so on. The heatsink is located at the bottom for removing generated heat from the chip. The vertical links are designed with TSVs which play the most important role for high-speed inter-die communication. On the side, the cross section of a 3D IC and TSV (Figure is not drawn to scale).](image-url)
integration technologies, through-silicon-via (TSV) has been most popularly adopted at the state-of-the-art 3D ICs [91] [92]. TSVs, in a 3D IC, connect different circuit and network components through the bonding pads and/or micro-bumps from adjacent dies and form the inter-die communication backbone, as shown in Figure 3.1.

Network-on-chip (NoC) is an emerging paradigm to revolutionize the manycore chip architecture. It is on the on-chip communication networks among the nodes (cores and routers), which decouples the communication fabric from the processing storage elements [93] [94] [95]. NoC exploits the benefits of parallelism and scalability at the same time.

Three-dimensional Network-on-Chip (3D NoC) combines the advantages aforementioned 3D integration and network-on-chip architectures to enable high performance manycore chip design [92] [53] [7]. It allows for the creation of new structures that enable significant performance enhancements over more traditional solutions. With freedom in the third dimension, architectures that were impossible or prohibitive due to wiring constraints in planar ICs are now possible, and many 3D implementations can outperform their 2D counterparts. Among the conventional wire-based 3D NoC architectures, 3D MESH, 3D Concentrated MESH, 3D MESH-BUS hybrid architecture, 3D BFT (butterfly) have been explored in the literature [7] [96]. Figure 3.2 shows these different NoC configurations.

### 3.2 Problem Description of 3D NoC Design

The goal of an on-chip communication system design is to transmit data with low latencies and high throughput using the least possible power and resources. In this context, design of small-world (SW) network-based NoC architectures [14] is a notable example. It has been shown that
either by inserting long-range shortcuts in a regular mesh architecture to induce a SW effect or by adopting a power-law based SW connectivity, it is possible to achieve significant performance gain and lower energy consumption compared to traditional multi-hop mesh networks [97] [7] [98]. In this work, we advocate that the concept of small-worldness should be adopted in 3D NoCs too. Specifically, the VLs in 3D NoC should enable the design of long-range shortcuts necessary for a SW network. However, the appropriate placement of the planar and the long-range links along the vertical dimension is crucial for maximizing the performance benefits. Hence, our goal is to optimize the placement of the planar and the vertical links in a 3D NoC where the overall interconnection architecture follows the small-world connectivity and improves the network latency and power consumption per message.

3.2.1 Small-world Network

An SW network lies in-between a regular, locally interconnected mesh network and a completely random Erdős-Rényi topology [99] [100]. SW graphs have a very short average path length, defined as the number of hops between any pair of nodes. The average shortest path length of SW graphs is bounded by a polynomial in log \(N\), where \(N\) refers to the number of nodes; this property makes SW graphs particularly interesting for efficient communication with minimal resources [15].

Figure 3.2 Different kinds of popularly used 3D NoC architectures with TSVs acting as the vertical links. From left, three NoC architectures are 3D MESH, Concentrated or Ciliated 3D MESH, and 3D MESH-BUS hybrid NoC. The planar NoC configuration is same for all while the vertical link design vary depending on the NoC configuration.
To develop small-world network, we follow the power-law based connectivity \cite{102} \cite{103} \cite{104}. The probability ($p$) of having a direct link between nodes in a SW network varies exponentially with the link length ($\ell$), i.e., $p(\ell) \propto \ell^{-\alpha}$ where, the parameter $\alpha$ governs the nature of connectivity, e.g., a larger $\alpha$ means a locally connected network with a few, or even no long-range links. By the same token, a zero value of $\alpha$ generates an ideal SW network following the Watts-Strogatz model \cite{99} \cite{100} – one with long-range shortcuts that are virtually independent of the distance between the cores. Figure 3.3 illustrates the small-world network which lies in between a completely regular and random network.

\begin{figure}[h]
\centering
\includegraphics[width=0.7\textwidth]{sw_network.png}
\caption{Illustration of small-world (SW) network. (Left) regular network, (middle) small-world (right) complete random network. The parameter $p$ refers to amount of randomness in the network. For regular network $p = 0$, while $p = 1$ for complete random network. In a SW network, there are only few long-range links which induce finite randomness ($0 < p < 1$) \cite{99}.}
\end{figure}

Our goal is to use the small-world (SW) approach to build a highly efficient 3D-NoC with planar links and TSVs in the vertical dimension. We consider 4 layers in this 3D NoC topology and the dimension of each layer is 10 mm x 10 mm.

### 3.2.2 Exploiting Small-world Networks for NoC Architectures

In the proposed 3D NoC topology (Figure 3.1), each core is connected to a switch; the switches are interconnected using planar and TSV links. The topology of the NoC is a small-world network where the links between switches are established following a power law distribution. More
precisely, the probability $P(i,j)$ of establishing a link between two switches $i$ and $j$, separated by an Euclidean distance $\ell_{ij}$, is proportional to the distance raised to a finite power as in [103]:

$$P(i,j) = \frac{\ell_{ij}^{-\alpha} f_{ij}}{\sum_{\forall i} \sum_{\forall j} \ell_{ij}^{-\alpha} f_{ij}}$$

(3.1)

The frequency of traffic interaction between cores, $f_{ij}$, is also factored in, so that the more frequently communicating cores have a higher probability of having a direct link between them. This frequency is expressed as the percentage of traffic generated from core $i$ that is sent to core $j$.

This approach implicitly optimizes the network architecture for a non-uniform traffic scenario. Getting now into details, the parameter $\alpha$ governs the nature of connectivity; in particular, a larger $\alpha$ would mean a locally connected network with a few, or even no long-range links. By the same token, a zero value of $\alpha$ would generate an ideal small-world network following the Watts-Strogatz model [99] [100] – one with long-range shortcuts that are virtually independent of the distance between the cores. It has been shown that $\alpha$ being less than $D + 1$, $D$ being the dimension of the network, ensures the small-worldness [99] [100]. Overall, the parameter $\alpha$, affects the NoC performance significantly. Thus, for our proposed architecture, we first focus on determining the parameter ‘$\alpha$’ considering the constraints of the 3D network structure. To be specific, as we consider TSVs for vertical connection, these links need to be perfectly aligned along the z-dimension and we can apply irregularities following power law-based interconnection in the planar dies only. Figure 3.4 shows an example of small-world-enabled 3D NoC (3D SWNoC) with four tiers and TSVs acting as the vertical links in between the tiers.

### 3.2.3 Development of 3D SWNoC

Starting from a power-law based connectivity, we attempt to optimize the location of the planar links and the VLs to achieve lower latency and energy consumption. We define an objective
function $O$ called communication cost, which combines the NoC performance metrics, namely the network latency and energy consumption per message. Optimizing the communication cost ensures lower average hop count and improvement in the network performance in terms of both latency and energy consumption. However, the space of physically feasible SW based 3D NoC designs $D$ is combinatorial in nature and our goal is to find the design $d \in D$ that minimizes $O$. One could employ search algorithms such as hill-climbing and simulated annealing, which are very popular in the design community for this task. However, we leverage machine-learning techniques that have been shown to improve the performance of these search algorithms by intelligently exploring the design space [105] [106]. This optimization process is undertaken before the actual NoC implementation.

3.2.4 Communication Cost function, $O$

In order to determine $\alpha$, we focus on optimizing the parameter, $O$, which is the average path length for any message. Optimizing the average path length ensures less utilization of network resources and improvement in the network performance both in terms of latency and energy dissipation. We define $O$ as the product of hop count, frequency of communication and link length between any pair of source and destination-

![Figure 3.4 Illustration of small-world network-based 3D NoC (3D SWNoC). Planar layers are designed with conventional wires while the vertical links are designed with through-silicon-vias (TSVs). In 3D SWNoC, TSVs act as long-range shortcuts necessary for inducing small-world effects.](image-url)
\[ O = \sum_{i} \sum_{j} (m \times h_{ij} + d_{ij}) \times f_{ij} \]  

(3.2)

Where \( h_{ij} \) is hop count between switches \( i \) and \( j \); \( m \) is number of stages inside a NoC switch; \( f_{ij} \) is frequency of communication and \( d_{ij} \) is physical distance corresponding to each hop calculated along the path. So, any network that possesses low \( O \) will achieve low message latency and energy dissipation and hence low energy-delay-product (EDP).

### 3.2.5 Network Constraints

To explore only physically feasible 3D NoC designs, we enforce some constraints on the placement of vertical links and router configurations. If TSVs are considered as the vertical links, we only allow placing them point-to-point (regularly) between the routers. Such constraints may put additional limits on the performance of NoC designs. However, efficient optimization can overcome such limitations. The SW network has an irregular connectivity. Hence, the number of links connected to each router is not constant. For fair comparison between our SW network and 3D MESH, we assume that both of them use the same average number of connections, \(<k_{avg}>\) per router. This also ensures that the 3D SW NoC does not introduce additional links compared to a 3D MESH. For a 64-core system, \(<k_{avg}>\) is 4.5 considering all the routers, including the peripheral ones. In addition, the maximum connectivity per node, \(<k_{max}>\), is set to be 7 for the SW network as found in [107].

### 3.3 Optimization of 3D SWNoC via Simulated Annealing

As mentioned above, the parameter \( \alpha \) governs the nature of the connectivity. Hence, we first determine the value of this parameter that will help us in designing the 3D SW NoC architecture with least EDP. To build 3D SW NoCs with optimum value of \( \alpha \), we develop the design flow
shown in Figure 3.5. The inputs to the flow are the total number of links, the total number of nodes, the locations of the cores, and $\alpha$.

### 3.3.1 Design and Optimization Steps

The design flow and simulated annealing-based optimization approach are shown in Figure 3.5. The description is as follows:

1. For a given $\alpha$, we calculate the link length distribution following (1). In this work, we constrain the number of total links equal to that of 3D MESH and the system size is 4x4x4.

2. We follow the constraint of perfectly aligned regular vertical link placement along the z-dimension. This is because, we use TSVs and they need to be perfectly aligned.

3. The small-world network has an irregular connectivity. Hence, the number of links connected to each switch is not a constant. For fair comparison between our small-world network and 3D MESH, we assume that both of them use the same average number of connections, $<k_{avg}>$ per switch. This also ensures that the 3D SW NoC does not introduce additional links compared to 3D MESH. For a 64-core system, $<k_{avg}>$ is 4.5 considering all the switches including the peripheral ones. In addition, the maximum connectivity per node, $<k_{max}>$, is set to be 7 for the SW network as found in [107].

4. To develop our SW network, we consider the communication frequency $f_{ij}$ between any pair of source and destination nodes and try to optimize $O$ for network performance. We map the tasks among the cores in such a way that the overall $O$ decreases. To do this, we place highly communicating and long distant nodes along the vertical dimensions. Placing the physically long links in the vertical dimension provides two benefits. The wireline energy consumption reduces
due to the reduction of the link length and the number of repeaters needed in a long planar wire. Moreover, the reduced network latency minimizing the probability of traffic congestion eventually reduces the overall power consumption.

5. After task remapping, we first build a random network with the link distribution determined at step 1. Then simulated annealing (SA) is performed to reduce $O$. A solution perturbation method we use in the simulated annealing is to randomly choose an existing link, remove the link, and create a new link of the same length between two other nodes. The convergence criteria are the total number of perturbations, the lowest temperature, and the lowest bound (0.1% in our experiments) on the amount of performance improvement compared to the previous network.

Figure 3.5 Simulated annealing-based 3D SWNoC design and optimization steps for achieving the best parametric values for network connectivity parameter, $\alpha$. 

Choose 3D SWNoC that gives lowest EDP
configuration after a pre-determined number of perturbations. In this way, we develop the optimum 3D SW NoC configuration for a given $\alpha$.

6. We build the optimum 3D NoC using the above steps for a given $\alpha$. Consequently, we vary $\alpha$ within a certain range (determined experimentally in section 4) and determine the optimum 3D NoC for each $\alpha$ using our design flow. Finally, to choose the optimum value of $\alpha$ for 3D SW NoC, we consider the particular network configuration that minimizes the EDP of the NoC. The simulation results will be shown in Section 3.7.

Following the above-mentioned steps and considering the constraints of network resources, we can develop optimized 3D SW NoC for any given set of applications.

### 3.4 Optimization of 3D SWNoC Via Machine Learning

In this section, we discuss the details of machine learning (ML)-based NoC optimization techniques.

#### 3.4.1 NoC optimization via ML: STAGE-based Approach

We employ an online learning algorithm called STAGE [105], which was originally developed to improve the performance of local search algorithms (e.g., hill climbing) with random-restarts for combinatorial optimization problems. The key insight behind STAGE is to leverage some extra features $\phi(d) \in \mathbb{R}^m$ ($m$ is the number of features) of the optimization problem to learn an improved evaluation function $E$ that can estimate the promise of a design $d$ as a starting point for the local search procedure $A$. It employs $E$ to intelligently select promising starting states that will guide $A$ towards significantly better solutions. Past work in the search community concluded that many practical optimization problems exhibit a “globally convex” or “big valley” structure, where the
set of local optima appear convex with one global optimum in the center [106] [108]. The main advantage of STAGE over popular algorithms such as simulated annealing is that it tries to learn the solution space structure and uses this information cleverly to explore a much bigger design space in the given time. To the best our knowledge, this is the first work that applies STAGE to an NoC design optimization problem.

The algorithm repeatedly alternates between two types of search as shown in Figure 3.6: 1) Base search, where $A$ is run with the original objective $O$ until it reaches a local optima and new training data is generated to improve $E$; and 2) Meta search, where it performs search with the learned evaluation function $E$ to select good starting states to improve the performance of the local search procedure $A$. We want to learn $E$ such that the estimated value of design $d$ is equal to the expected best objective ($O$) value seen on a search trajectory that starts from design $d$ and follows the local search method $A$ guided by $O$. In the initial exploration phase, $E$ may not lead to good solutions but as the iterations progress, $E$ will improve with the training data generated from the search experience in base search mode. The effectiveness of the learned $E$ depends on a small subset of
critical training examples that successfully teach how to avoid different local optima during the meta-search phase. The STAGE algorithm tries to quickly identify this critical set in an adaptive manner.

We initialize $E$, training set $Z$, and initial design $d_0$. The following high-level algorithmic steps of STAGE are repeated for several iterations.

### 3.4.1.1 Base search using $A$ guided by $O$

From $d_0$, run the search procedure $A$ until a local optima is reached thereby leading to a search trajectory $(d_0, d_1, ..., d_T)$. In general, the base search is a local search algorithm e.g. greedy search, simulated annealing, stochastic gradient descent and so on.

### 3.4.1.2 Improve $E$

For each design $d_i$ on the search path, add $(\phi(d_i), y_i)$ to $Z$, where $y_i$ is the best value along the search. Re-train $E$ using a regression learner $R$ with the updated training set $Z$.

### 3.4.1.3 Meta search guided by $E$

Continue from $d_T$ and optimize $E$ by performing a hill-climbing search to produce the best predicted starting state $\hat{d}$. If $\hat{d}$ is the same as $d_T$ (no search progress), set $d_0$ to a random design. Otherwise, set $d_0 = \hat{d}$.

At the end, we return the best design found over all the iterations.

### 3.4.2 Instantiation for 3D NoC Optimization for ML-based Approach

In this section, we provide all the details needed to apply the STAGE algorithm to our 3D NoC optimization problem.
3.4.2.1 Design Space for ML-based Approach

Our design space depends on a set of network resources, which are given as input to the optimization algorithm. These resources are defined as follows. 1) Cores ($C$): A set of all cores $C = \{C_1, C_2, ..., C_N\}$, where $N$ is total number of cores. We assume that every core is connected to at least one router; 2) Planar Dies ($P$): A set of all dies $P$. For $N = 64$, we consider four dies with each die containing 16 cores. For core placement, we follow a greedy algorithm to minimize $(f_{ij} \cdot d_{ij})$, where $f_{ij}$ and $d_{ij}$ are the communication frequency and Cartesian distance between the cores respectively. In this step, we form clusters with 16 cores in each die; 3) Link Distribution ($L$): The link length distribution $L = \{l_1, l_2, ..., l_k\}$, where $k$ depends on the size and topology of the network; $l_i$'s are determined based on the SW connectivity parameter $\alpha$. For higher values of $\alpha$, $l_k$ decreases; and 4) Communication Frequency ($F$): The communication frequency among different cores $F = \{f_{ij} \mid 1 \leq i, j \leq N, i \neq j\}$. We assume that $F$ for each application is given as an input to perform application-specific network optimization.

![Figure 3.7](image.png) Instantiation of STAGE-based optimization algorithm for 3D SWNoC link placement. Two searches viz. the base search, $A$, and meta search, $M$, iterate in between them by seeking guidance from the evaluation function, $E$, to reach to the optimized NoC architecture quickly.
The set of all physically realizable SW NoC designs with the given link distribution $L$ forms our design space.

We utilize the same objective function, $O$ as the communication cost as defined earlier in Section 3.2.3 for simulated annealing-based approach.

---

**Algorithm 3.1: NoC Design Optimization via STAGE Algorithm**

1: **Input**: $D$ = Design space,  
   $O$ = cost function,  
   $(I,S)$ = initial state and successor generation functions,  
   $C$ = network constraints,  
   $\phi$ = feature function for NoC design,  
   $A$ = local search procedure,  
   $R$ = regression learner,  
   $\text{MAX}$ = maximum iterations,  

2: **Output**: $d_{\text{best}}$, the best NoC design  

3: **Initialization**: initialize evaluation function $E$, training set $Z$, initial design $d_0$,  
   $O_{\text{best}} = O(d_0)$, and $d_{\text{best}} = d_0$  

4: **Repeat**:  
5: **Base search**: From $d_0$, run the search procedure $A$ guided by $O$ until a local optima is reached, leading to a search trajectory $(d_0, d_1, ..., d_T)$.  

6: **Generate training data**: For each design $d_i$ on the search trajectory, add $(\phi(d_i), y_i)$ to $Z$, where $y_i$ is the best value along the search trajectory.  

7: **Re-train $E$**: $E = R(Z)$.  

8: **Meta search**: From $d_T$, run the search procedure $A$ guided by $E$ until a local optima is reached to produce the best predicted starting state $\hat{d}$.  

9: **Next starting state**: If $\hat{d} = d_T$ (no search progress), set $d_0$ using $I$. Otherwise, set $d_0 = \hat{d}$.  

10: **Update $O_{\text{best}}$ and $d_{\text{best}}$** if $y^* < O_{\text{best}}$, where $y^*$ is the best value encountered during base search and meta search.  

11: **Until** $\text{MAX}$ iterations or convergence.  

12: **Return** best design $d_{\text{best}}$.  

29
3.4.2.2 Starting States and Successor Function

For starting states, we randomly generate a SW network that satisfies the network constraints. The successor function $S$ takes a network as input and returns a set of next states and allows the search procedure to navigate the NoC design space. $S$ generates one candidate state for each link connecting two nodes in the input network. It simply removes that link and places a link with the same length between two nodes in the network that are not directly connected.

The STAGE algorithm can benefit if we can specify the starting state distribution using some domain knowledge. Therefore, we also consider a starting-state distribution named $\alpha$-Greedy. We formulate the starting state (design) construction as a sequential decision-making task, where we select the next link to be placed at each step. In $\alpha$-Greedy distribution, we select a link greedily with probability $\alpha$ based on communication frequency and a random link with probability $(1-\alpha)$. We start with $\alpha=1$ (completely greedy) and gradually reduce $\alpha$ to increase the randomness.

3.4.2.3 Local Search Procedure $A$

We employed a stochastic hill-climbing procedure, where the next states are sampled stochastically. More specifically, from the current state of the NoC, we randomly pick a neighborhood NoC. Call a successor function and perturb the solution. We randomly pick a link and remove followed by replacing the same link with another pair of nodes. The cost functions are evaluated to determine to select those randomly picked nodes (randomly generated 3D NoC configurations) to be selected or not as the current solution.
3.4.2.4 Feature Function $\phi$

The main challenge in adapting STAGE to our NoC domain is to define a set of features $\phi$ for each network that can drive the learner. We divide the whole network into several overlapping subgraphs or regions, and define a set of features that can be categorized into three types: 1) Average hop count ($h$), which calculates the average hop count for each region or sub-network; 2) Weighted communication which is defined as the sum of the products of hop count and communication frequency over all source-destination pairs for a particular hop count ($\sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} f_{ij} \times h_{k}$). The highest value of $k$ depends on the network size and topology. If the value of this feature is small, it indicates that highly communicating cores are placed in the same neighborhood; and 3) Clustering coefficient ($C_c$), which captures the connectivity of one core with its neighbors [109]. While the hop count takes into account mainly long-range communication, the clustering coefficient focuses more on local connectivity among the immediate neighbors. We found these features to sufficiently capture the network characteristics, efficient to compute, and allow to learn highly accurate evaluation function $E$. Table 3.1 shows the description of feature vectors adopted for this work.

Table 3.1: Description of feature vectors used for SWNoC optimization

<table>
<thead>
<tr>
<th>Feature Type</th>
<th>Feature Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. hop count for nine overlapping regions</td>
<td>9</td>
</tr>
<tr>
<td>Weighted communication ($\sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} f_{ij} \times h_{k}$) considering maximum hop count for $N=64$</td>
<td>8</td>
</tr>
<tr>
<td>Avg. Clustering coefficient ($C_c$) for four planar dies</td>
<td>4</td>
</tr>
</tbody>
</table>
3.4.2.5 Regression Learner

The quality of our optimization methodology depends on the accuracy of the evaluation function, $E$. We can employ any regression learning algorithm e.g. $k$-nearest-neighbor ($k$-NN) [110], linear regression (LR) [111], support vector regression (SVR) [112] [113], and regression tree (RT) [114] [105].

3.4.2.5.1 Support Vector Regression (SVR) Learner

Our training data consists of a set of input-output pairs $\{(x_i, y_i)\}_{i=1}^n$, where each $x_i \in \mathbb{R}^m$ is a feature vector and $y_i \in \mathbb{R}$ is the corresponding output. The $\varepsilon$-SVR algorithm tries to learn a function $E$ such that the deviation of the predicted output $E(x_i)$ from correct output $y_i$ is less than the error tolerance parameter $\varepsilon$.

Without loss of generality, we assume $E$ is a linear function of the form $E = \langle w, x \rangle + b$, and present the $\varepsilon$-SVR formulation in primal form:

$$\begin{align*}
\min & \quad \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \\
\text{s.t.} & \quad y_i - \langle w, x_i \rangle - b \leq \varepsilon + \xi_i \\
& \quad \langle w, x_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \\
& \quad \xi_i, \xi_i^* \geq 0
\end{align*}$$

In supervised learning, the goal is to learn a function that will perform well on unseen examples (generalization) and not the one that minimizes the error on the training data (over-fitting). This problem is generally addressed by adding a penalty term to discourage complex functions. In Equation (3.3), the first part is the penalty term (regularizer) and the second part is the training error. $C$ is the regularization parameter that provides the tradeoff between minimizing the training...
error and generalization to unseen data, and $\zeta_i$ and $\xi_i^*$ are slack variables to handle infeasible constraints. Linear functions won't suffice for complex problems such as ours. Therefore, we employ the radial basis function (RBF) kernel ($K(x, x') = e^{\gamma \|x-x'\|^2}$) to be able to learn non-linear functions, where $\gamma$ is a tuning parameter. We selected the RBF kernel over other kernels because of its flexibility and predictive power. Additionally, to find the best learned function, we need to search over different values of $(\varepsilon, C, \gamma)$. We employ LibSVM [115] [116] to learn the regression function and select the best combination of $(\varepsilon, C, \gamma)$ via the inbuilt v-fold cross-validation approach.

To this end, a regression learner that is non-linear, fast in terms of training time and prediction time will improve the effectiveness of the STAGE algorithm. Therefore, analytically, the regression tree learner suits our needs the best. Hence, in order to achieve faster convergence of the optimization algorithm, we employ a regression tree-based learner for the rest of the work.

### 3.4.2.5.2 Regression Tree-based Learner

Our training data consists of a set of input-output pairs $\{(x_i, y_i)\}_{i=1}^n$, where each $x_i \in \mathbb{R}^m$ is a feature vector and $y_i \in \mathbb{R}$ is the corresponding output. The regression tree learning algorithm tries to learn a function $E$ in the form of tree (a set of if-then rules) to minimize the deviation of the predicted output $E(x_i)$ from the correct output $y_i$. The key idea in regression tree learning is to recursively-partition the input space (as in hierarchical clustering) until we find regions that have very similar output values. The recursive partitioning is represented as a tree, where leaves correspond to the cells of the partition. Each leaf is assigned the sample mean of all the output variables in that cell as its prediction. During testing, we find the cell of the partition that input $x$ belongs to through a series of comparison questions on the features, and return the prediction associated with that cell. Regression trees also allow us to identify the features that are important in making predictions.
We employed the WEKA machine-learning toolkit [117] [118] to train regression trees over training set $Z$, and tune the hyper-parameters using validation data.

### 3.5 Irregular 3D NoC Topologies under Consideration

To benchmark the performance of our proposed 3D SW NoC architecture with respect to other irregular 3D NoCs, we consider the Mesh-Random-Random-Mesh (mrrm) and Random-random-random-random ($rrrr$) architecture as suggested in [57]. Like the 3D SW architecture, in this case also we keep the total number of links to be equal to that of a 3D MESH.

The characteristics of these two architectures are:

- **mrrm**: Point to point (P2P) TSV based 3D NoC that consists of two planar layers of mesh-based interconnection architecture and the rest two layers have random interconnection patterns.

- **rrrr**: P2P TSV based 3D NoC that consists of four layers of random interconnection networks.

To construct the *random* architectures for *mrrm* and *rrrr* configurations, 1000 random connection matrices were generated and we obtained the average of the average hop count over these 1000 matrices. Then the particular connection matrix, which has the closest average hop-count to the average of average hop-count, was selected for the *random* architecture.

### 3.6 Routing Algorithm

The performance of NoC largely depends on the routing strategy adopted. In this case, we use incorporate suitable routing strategies depending on the NoC topology.
3.6.1 Adaptive Layered Shortest Path Routing (ALASH)

For regular 3D MESH architecture, XYZ or adaptive-Z are the preferred routing algorithms for their simplicity. For irregular architectures such as the small-world network, the topology agnostic Adaptive Layered Shortest Path Routing (ALASH) algorithm is proved to be suitable [119]. ALASH is built upon the layered shortest path (LASH) algorithm but has better flexibility by allowing each message to adaptively switch paths, letting the message choose its own route at every intermediate switch. We incorporate the ALASH routing for our 3D SW architecture.

3.6.2 Dimension-ordered Routing for MESH NoCs

For regular 3D MESH architecture, dimension-ordered routing XYZ or adaptive-Z are adopted as the routing algorithms for their simplicity [120] [121].

3.7 Experimental Results and Analysis

In this section, we first present the achievable performance and energy consumption profiles of our optimized 3D SW NoC architecture. Then we present a detailed reliability analysis in the presence of spare-VL insertion. For this performance evaluation, we consider three metrics: latency, energy consumption, and energy-delay-product (EDP) per message. The EDP is defined as the product of network latency and energy consumption, and it unifies both of them into a single parameter.

3.7.1 Experimental Setup

To evaluate the performance of different NoCs, we use a cycle-accurate NoC simulator that can simulate any regular or irregular 3D architecture [107] [7] [102] [103]. Our system consists of 64 cores and 64 network routers equally partitioned in four layers. The length of each packet is 64
flits and each flit consist of 32 bits. The routers are synthesized from an RTL level design using TSMC 65-nm CMOS process in Synopsys™ Design Vision. All router ports have a buffer depth of two flits and each router port has four virtual channels in case of irregular NoC. The NoC simulator uses wormhole routing, where the data flits follow the header flits once the router establishes a path. For regular 3D mesh-based NoC, XYZ-dimension order-based routing is used. For irregular architectures such as the SW network, the topology-agnostic Adaptive Layered Shortest Path Routing (ALASH) algorithm is adopted [119]. The energy consumption of the network routers, inclusive of the routing strategies, was obtained from the synthesized netlist by running Synopsys™ Prime Power, while the energy dissipated by wireline links was obtained through HSPICE simulations, taking into consideration the length of the wireline links. We consider four SPLASH-2 benchmarks, namely, FFT, RADIX, LU, and WATER [122], and five PARSEC benchmarks, namely, DEDUP, VIPS, FLUIDANIMATE, CANNEAL, and BODYTRACK (BT) [123] in this performance evaluation. These benchmarks vary in characteristics from computation intensive to communication intensive in nature and thus are of particular interest in this work.

Figure 3.8 Variation of normalized EDP with respect to the small-world connectivity parameter $\alpha$. The EDP is normalized with respect to the 3D MESH for each respective benchmark.
3.7.2 Determination of Connectivity Parameter, $\alpha$ and Task Remapping

In this section, we first determine the connectivity parameter $\alpha$ for our proposed 3D SW NoC. We vary the connectivity parameter $\alpha$ from 1.0 to 3.6 and find the best $\alpha$ that gives us the lowest energy-delay product (EDP). Figure 3.8 shows the EDP values of our 3D SW NoC architecture for each $\alpha$. The EDP values are normalized to those of 3D MESH NoC for all the benchmarks. As seen from the figure, as $\alpha$ increases from 1.0 to 2.4, the EDP decreases almost steadily. When $\alpha$ is small, large number of long-range links are likely to be placed at the cost of reduced local, short-range links. Although this configuration improves long-distance data exchange, it compromises the local communications, which have larger impact on the EDP than the long-distance communications. Thus, the EDP is high when $\alpha$ is small. As $\alpha$ increases from 2.4 to 3.6, most of the links become local, short-range links and the network approaches the regular multi-hop pattern that also results in high EDP. Therefore, we need to choose a value of $\alpha$ that achieves a compromise between these two extreme cases. If we increase $\alpha$ beyond 3.6, the long-range links become almost non-existent and the NoC architecture becomes very close to 3D MESH. Hence, we do not consider $\alpha$ beyond this.

![Figure 3.9 Average normalized EDP of 9 benchmarks of 3D SWNoC with variation in connectivity parameter, $\alpha$ (normalized to 3D MESH).](image)
To visualize the effect of $\alpha$ more clearly, Figure 3.9 plots the average EDP of the nine benchmarks. As shown in the figure, $\alpha=2.4$ gives us the lowest EDP on average, so we choose $\alpha=2.4$ for our 3D SW NoC design.

Remapping tasks among the cores helps to improve NoC performance. However, task remapping may affect the traffic congestion. Figure 3.10 shows the percentage of total traffic carried by each die before and after remapping the tasks. We observe from this figure that the task remapping slightly increases the traffic carried by the second and the third dies, but it reduces the traffic in the first and the fourth dies. Therefore, it is clear that our task remapping does not lead to noticeable traffic congestion in any particular layer.

### 3.7.3 Performance of the Optimization Algorithm

In Section IV, we described the details of the STAGE optimization algorithm for designing the 3D SWNoC architecture. Here we first characterize the performance of the optimization algorithm by quantifying various performance metrics of the optimized 3D SWNoC. To evaluate the performance of STAGE algorithm, we compare it with the well-known combinatorial optimization

![Figure 3.10 Percentage of total traffic carried out by each layer (Die 1 to 4) before and after task remapping.](image)

38
algorithm, viz., simulated annealing (SA) [124] [125]. We evaluate the performance in terms of both the quality of solution and their individual convergence time.

3.7.3.1 Communication Cost \((O)\) and Prediction Error of STAGE-based Approach

We create the initial network following the power law distribution shown in Section 3, where long-range links are placed randomly. Our goal is to find an optimized network starting from this random SW network. We call this initial NoC architecture as 3D SW_random. Figure 3.11 shows the communication cost of the optimized network as a function of the number of iterations of the STAGE algorithm and the corresponding prediction error (in %) of the learned function. During this process, the learned function \(E\) predicts an initial network configuration to start the local search procedure that can lead to lower communication cost \((O)\). As seen from the figure, during the initial exploration phase, the error-rate is non-monotonic and high. After a few iterations the prediction error reduces to less than 1%, and after 20 iterations, the error is almost zero (0.05%). The prediction error remained more or less the same for all the subsequent iterations. These results indicate the effectiveness of our network features \(\phi\) and the SVR learning algorithm. Note that the best \(O\)-value decreases monotonically as the set of explored designs increases over the iterations.

![Figure 3.11 Best objective value \(O\) and error rate of the evaluation function \(E\) for the STAGE algorithm over iterations.](image-url)
We also ran the same experiment with the $\alpha$-Greedy starting state distribution as mentioned above. However, the communication cost $O$ and the prediction error have similar characteristics as the random distribution for the benchmarks and the system size considered in this work. Therefore, we present and discuss our results with random starting-state distribution.

The learned evaluation function $E$ becomes highly accurate after a small number of iterations and produces good starting states to help the local search procedure $A$ in producing optimized network architectures with lower objective $O$ (communication cost). We denote the final optimized NoC as **3D SW\_opt**.

### 3.7.3.2 STAGE vs. Simulated Annealing (SA)

We create the initial network following the power law distribution shown in Section 3.2, where long-range links are placed randomly. Our goal is to find an optimized network starting from this random SW network. We call this initial NoC architecture as **3D SW\_rand**. Figure 3.12 shows the communication cost of the optimized network from the STAGE and SA algorithm as a function of time.

![Figure 3.12](image)

**Figure 3.12** Performance comparison between the machine-learning-based optimization algorithm (STAGE) and the Simulated Annealing (SA) algorithm.
To compare the performance of these two optimization algorithms, we consider two parameters, viz., the quality of the solution and the convergence time. To make the comparison fair, we consider the same NoC configuration and apply both STAGE and SA algorithm to optimize it. We used a machine configured with Intel Core i7-4700MQ processor and 8 GB RAM running at a clock frequency of 2.4 GHz.

Figure 3.12 shows the cost of the best solution obtained at any particular time for SA and STAGE. We consider the best explored cost, $O_{\text{best}}$, as the quality of the optimization algorithm. It is evident that STAGE reaches $O_{\text{best}}$ very fast (within 5 minutes). During the optimization process, the learned function $E$ predicts an initial network configuration to start the local search procedure that can lead to lower communication cost ($O$). During the initial exploration phase, the error-rate is non-monotonic and high. After a few iterations the prediction error reduces to less than 1%, and after 20 iterations, the error is almost zero (0.05%). The prediction error remained more or less the same for all the subsequent iterations. These results indicate the effectiveness of our network features $\phi$ and the regression-learning algorithm. Note that the best $O$-value decreases monotonically as the set of explored designs increases over the iterations. We also ran the same experiment with the $\gamma$-Greedy starting state distribution as mentioned above. However, the communication cost $O$ and the prediction error have similar characteristics as the random distribution for the benchmarks and the system size considered in this work. Therefore, we present and discuss our results with a random starting-state distribution.

It is also seen that, SA reaches $O_{\text{best}}$ more gradually, and even after 50 minutes it does not reach the same solution as STAGE. It should be noted that we have to optimize the link locations for various applications. Hence, this additional time needed by SA will be a significant overhead when
we have to optimize and reconfigure the 3D SWNoC in the field. Hence, we can conclude that STAGE algorithm is more efficient in designing an optimized 3D SWNoC with better performance.

Figure 3.13 Effect of optimization algorithm on weighted communication ($w_{ij} = \text{sum of all} (f_{ij} \times h_{ij})$) features.

3.7.3.3 Characteristics of the Design: Random vs. Optimized

Now we investigate why the STAGE based optimization algorithm is suitable for developing energy-efficient NoC architectures. In Section 3.4.2, we described the details of the feature definition ($\phi$), to represent each network. So, we will explore how the design features change before and after the optimization process. Here we specifically consider the role of the weighted communication feature mentioned in Section 3.4.2. Figure 3.13 shows the weighted communication feature, which reveals the percentage of total communication that is constrained between two nodes separated by $k$ hops ($k \geq 1$). Careful observation of Figure 3.13 shows that for 3D SW_opt, the traffic constrained within one, two, and three hop increases compared to 3D SW_rand. Moreover, the amount of traffic that has to traverse beyond three hops decreases.

Hence, the inter-node communication that takes place in less than three hops becomes more frequent. Since the average hop count of the optimized network is calculated to be 2.94, any
communication below this average hop count can be considered to be efficient. Essentially, the optimized network becomes more efficient for the same objective function.

The inset in Figure 3.13 shows the percentage of communication versus the number of hops, where the area under the curve denotes the weighted communication feature mentioned in Section 3.4. We can see that the 3D SW_opt curve shifts towards the left, which means that on an average, any message in the optimized network traverses less hops compared to the initial network. Hence, it spends less time inside the network and occupies less network resources. Therefore, the STAGE-based optimization algorithm guides the search to converge to an efficient architecture.

3.7.4 Effects of Optimization on 3D SW NoC

In this section, we evaluate and compare the performances of the 3D SW_opt and the un-optimized 3D SW_random architectures. For comparison purpose, all the values are normalized with respect to 3D MESH.

![Normalized network latency per message of 3D SWNoC before and after optimization.](image)

3.7.4.1 Network Latency

Figure 3.14 demonstrates the effect of optimization for 3D SW NoC. The optimization improves the network latency on an average of 3% over the un-optimized version, and 5.5% over the
conventional 3D MESH. The optimization process redistributes the links among the cores such that cores that have to frequently communicate with each other are either directly connected or need to traverse a small number of hops. This results in reduced average hop count and weighted communication for 3D SW_opt NoC. A new set of benchmarks with higher injection rates will highlight the benefits of this work even more, which is the focus of our future work.

![Normalized energy consumption per message of 3D SWNoC before and after optimization.](image)

**3.7.4.2 Energy Consumption**

Energy consumption per message depends on the energy consumed by the router as well as the planar and vertical links. The STAGE-based optimization algorithm reduces average hop count and communication cost, which contributes to the minimization of the router and link energy consumption respectively. Figure 3.15 plots the energy consumption profile before and after optimization normalized to these values for the 3D MESH. On an average, an optimized 3D SW NoC shows 33% and 17% energy consumption improvement over the 3D MESH and 3D

![Normalized EDP per message of 3D SWNoC before and after optimization.](image)
SW_random respectively. Figure 3 helps us in understanding the reasons behind the improvement in energy consumption. The area under the 3D SW_opt curve is less than that of the un-optimized counterpart. Hence 3D SW_opt reduces the utilization of network resources for any message. As a result, both the router and link energy decrease and hence, the overall energy consumption profile improves.

3.7.4.3 Energy-delay-Product (EDP)

From the EDP profile shown in Figure 3.16, we observe that the average EDP of 3D SW NoC is reduced by approximately 35% and 19% compared to 3D MESH and 3D SW_random respectively. The improvement in the energy-delay product is a direct consequence of the improvement in network latency and energy consumption of the 3D SW_optimized NoC.

<table>
<thead>
<tr>
<th>NoC architecture</th>
<th>Avg. hop count</th>
<th>Communication cost, $O$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D SW</td>
<td>2.94</td>
<td>69.45</td>
</tr>
<tr>
<td>mrrm</td>
<td>3.07</td>
<td>76.15</td>
</tr>
<tr>
<td>rrrr</td>
<td>3.03</td>
<td>76.34</td>
</tr>
<tr>
<td>3D MESH</td>
<td>3.81</td>
<td>83.47</td>
</tr>
</tbody>
</table>

Table 3.2: Comparison of Average hop count and communication cost of 3D NoC Architectures

3.7.5 Performance of 3D SWNoC Compared to Other 3D NoCs

In the previous sections, we showed that 3D SW_optimized significantly outperformed 3D SW_random. In this section, we compare 3D SW_optimized with several existing 3D NoC architectures. Henceforth, we refer 3D SW_optimized as 3D SW for simplicity. For this comparative performance evaluation, we consider 3D MESH and two recently proposed irregular 3D NoCs, namely, mrrm and rrrr [57]. Both the mrrm and rrrr NoCs have point-to-point vertical connections as in 3D MESH and 3D SW. However, their die-level planar connection pattern varies.
For \textit{rrrr}, all the four dies have randomly connected interconnection patterns. On the other hand, \textit{mrrm} has random connection patterns in the middle two dies whereas the first and the fourth dies follow mesh-based regular connectivity. To build \textit{mrrm} and \textit{rrrr}, we follow the method suggested in and keep the number of links equal to that of 3D MESH and 3D SW. All the performance metric values are normalized with respect to the 3D MESH.

### 3.7.5.1 Network Latency

Figure 3.17 shows the network latency of all the 3D NoCs. Among all the NoCs, 3D MESH and 3D SW exhibit the highest and the lowest latency respectively. The other two architectures namely \textit{mrrm} and \textit{rrrr} perform somewhere in the middle. As in the case of 3D SW NoC, both \textit{mrrm} and \textit{rrrr} have irregularities in the horizontal planes. However, the number and the length of the links are not optimized for these architectures. For \textit{rrrr}, the link distribution has large number of long-range links that help communication among long-distant cores at the expense of near-by communication. In the case of 3D SW NoC, the link distribution follows the power law and the connection pattern is optimized to facilitate both the nearby and long-range communications.

The \textit{mrrm} architecture maintains the link distribution in between \textit{rrrr} and 3D SW NoC. Hence, its network latency lies in between \textit{rrrr} and 3D SW. Finally, 3D MESH NoC suffers from higher average hop count compared to other 3D architectures due to multi-hop communication pattern;

![Normalized network latency of different 3D NoCs (3D MESH, mrrm, rrrr, and 3D SWNoC). All the values are normalized with respect to that of 3D MESH for respective benchmark.](image-url)
hence, it suffers from the highest network latency. Table 3.2 lists the communication costs and average hop counts for all these NoCs. As expected, 3D SW and 3D MESH exhibit the lowest and

Figure 3.18 Normalized energy consumption per message of different 3D NoCs (3D MESH, mrrm, rrrr, and 3D SWNoC). All the values are normalized with respect to that of 3D MESH for respective highest communication cost and hop count respectively, whereas mrrm and rrrr reside in between these two. The effect of these communication costs is eventually reflected in the latency characteristics.

3.7.5.2 Energy Consumption

Figure 3.18 shows the energy consumption per message for different 3D NoCs. Among all these, 3D MESH has highest energy consumption followed by mrrm, rrrr, and 3D SW NoC. Higher network latency gives rise to higher network resources utilization and hence, higher energy consumption per message. For 3D MESH, the router energy consumption is significantly higher due to multi hop communication, so it performs the worst among all of them. The mrrm and rrrr NoCs are capable of reducing the router energy consumption compared to mesh and performs

Figure 3.19 Normalized EDP per message of different 3D NoCs (3D MESH, mrrm, rrrr, and 3D SWNoC). All the values are normalized with respect to that of 3D MESH for respective benchmark.
better than 3D MESH. However, due to their random link distribution, they suffer from higher communication cost and average hop count compared to the optimized SW NoC. Hence, they consume more link energy and router energy. With the least communication cost, 3D SW NoC consumes the lowest energy possible among all these architectures.

3.7.5.3 Energy-delay-Product (EDP)

The energy-delay-product is directly affected by network latency and energy consumption. The architecture that performs best in terms of latency and energy consumption is expected to have lower EDP compared to the others. Figure 3.19 presents the EDP profile of different 3D NoCs. As expected, 3D SW NoC has the lowest EDP profile followed by mrrm, rrrr and 3D MESH. On an average, 3D SW has 35% lower EDP profile compared to 3D MESH while the highest improvement of 43% was found for RADIX.

3.7.6 Robustness of 3D SW NoC Architecture

In this section, we analyze the robustness of the 3D SW NoC architecture under vertical link failure. The reason behind addressing the scenario of vertical link failures scenario is that despite the recent advancements in the TSV technology, TSVs are still subject to failure due to voids, cracks, and misalignment [22] [61].

In this case, after building and optimizing the 3D SW NoC, the vertical links are randomly removed to simulate the link failure scenario. This principally tests the robustness of the SW interconnection network [15] [104]. Starting from the fault free 3D SW NoC, we increase the link failure percentage with a step size of 5% till 50% of the vertical links are randomly removed. In Figure 3.20, we show the average network latency, energy consumption, and energy-delay-product (EDP) for all the benchmarks by varying the amount of failed links. All parameters are again normalized.
to the values of fault-free fully connected 3D MESH. The figures show the worst, best and average performance levels for the same amount of link failures over 1000 different runs.

From these figures, we observe that as the link failure percentage increases, the network latency, energy consumption, and EDP increase gradually. The difference between the worst- and best-case scenarios also increases progressively. This occurs due to the fact that with increasing vertical link failure, we have fewer routing resources than what is required to achieve optimum performance. In addition, for the case of 50% vertical link failure, the average network latency and energy consumption almost equal the corresponding values for a fully connected fault-free 3D MESH. For this case, it is also quite striking that EDP matches with that of the fault-free 3D MESH as well. In the worst case, the network latency shows 2% higher value whereas, the energy

Figure 3.20 3D SW NoC performances normalized to fault free (fully vertical connected) 3D MESH Vs. the percentage of vertical link failure rate. (a) Average normalized network latency, (b) Average energy consumption per message, (c) Average EDP.
consumption and EDP record 13% and 15% higher values respectively compared to fault free 3D MESH. These results therefore show that even with significantly fewer number of vertical links, the 3D SW NoC performs as well as the fully connected fault free 3D MESH on average. It has been observed in [104] [15] that SW networks display remarkable resilience to high rates of link failures since the average distance between nodes in a SW network increases by a small margin with the rate of failures. Hence, the effect of vertical link failures on performance is minimal.

### 3.7.7 Optimization Quality with less Resources

We also analyze the quality of the proposed optimization algorithm in the presence of limited resources. To do so, we consider the 3D SW NoC with 50% vertical link failure and optimize the placement of the remaining links following our STAGE algorithm-based methodology. We show that the optimized NoC with reduced vertical links can still maintain a high-level of performance if the limited resources (vertical links in this case) are utilized optimally. For the rest of the work, we denote this optimized architecture as 3D SW\_partial and compare its performance with respect to the fully connected 3D MESH.

Figure 3.21 shows the latency, energy, and EDP for 3D SW\_partial with respect to the fully connected 3D MESH. We observe that on an average, 3D SW\_partial still shows 2.5% lower latency compared to the fully connected 3D MESH. In comparison to the results for the fully connected 3D SW NoC shown in Figure 3.14, 3D SW\_partial incurs only 3% higher network latency. The reason for this behavior is that the optimization algorithm ensures the most suitable link placement considering the available resources. As a result, the latency penalty remains low. Similarly, 3D SW\_partial shows 24% and 25% lower energy and EDP respectively compared to the fully connected 3D MESH. In addition, by comparing these energies and EDP values with
Figure 3.15 and Figure 3.16 for the fully connected 3D SW, we see that 3D SW_partial pays only 12% and 13% penalty respectively over its fully connected counterpart.

This result carries the promise of highly energy-efficient design with reduced resources. For 3D SW_partial NoC, we have reduced 50% of the vertical links, which are predominantly long-range shortcuts and still, on an average, we incur no more than 15% penalty. The algorithm optimizes the link distribution among the cores such that the overall communication cost is minimized. Hence, the 3D SW NoC with reduced number of vertical links utilizes its resources very efficiently to compensate for the resource reduction.

If we compare the performances of 3D SW NoC with 50% vertical link failure without any optimization (Section 3.7.6) with the optimized 3D SW_partial, then we find that the later performs better in every performance metric compared to the former. On an average, for latency, energy, and EDP metrics, 3D SW_partial shows improvements of 2.5%, 27%, and 26% respectively compared to its unoptimized counterpart. Hence, the optimization algorithm plays a crucial role in minimizing the performance penalty due to the limited resources. We can emphatically conclude that the proposed 3D SW NoC along with the optimization methodology is robust enough to compensate for the performance loss due to vertical link reduction. Overall, the performance penalty is small compared to the proportion of resource reduction.
3.7.8 Performance of 3D NoCs in Presence of Link Failures

In this section, we analyze the robustness of the 3D SWNoC architecture under VL failure. The reason behind studying the scenario of VL failures is that despite the recent advancements in TSV technology, TSVs are still subject to failure due to voids, cracks, and misalignment [23] [24]. In addition, TSVs also face the wear-out problem due to stress arising from potentially high workload. The imbalance in workload among different TSV-based VLs in the NoC also creates wide variation in TSV mean-time-to-failure (MTTF), where some VLs fail early compared to others. Due to all these reasons, if the TSV-based VLs fail, then the EDP and network latency of 3D NoC increases and in the worst case, the corresponding NoC may contain disjoint source-destination pairs in the network. As a result, the performance of the 3D NoC degrades over time.

Figure 3.22 demonstrates the EDP profile of 3D MESH, \textit{mrrm}, \textit{rrrr}, and 3D SWNoC with workload induced VL failure scenario with time for the CANNEAL benchmark. Just like the previous plots, all the EDP values are normalized with respect to fault free 3D MESH at t=0. From the figure, we can see that the EDP
values of all the 3D NoCs increase with time as the VLs fail progressively. Among all the NoCs, 3D SWNoC shows the lowest EDP value for any particular period of time and the rate of increase in EDP is also lower than the other NoCs. As a result, 3D SWNoC is expected to have longer lifetimes relative to other NoCs. Note that 3D SWNoC is inherently robust against link failure and its average hop count increases only marginally in the presence of link failures due to the small-world nature of the overall connectivity [101] [15]. As a result, it shows better robustness and EDP profile in comparison to other NoCs.

3.8 Summary of Small-world based 3D NoC Design

We proposed a robust design optimization methodology to improve the energy efficiency of 3D NoC architectures by combining the benefits of SW networks and machine learning techniques to intelligently explore the design space. We showed that the optimized 3D SW NoC architecture outperforms existing 3D NoCs. The optimized 3D SW NoC on an average achieves 35% EDP reduction over conventional 3D MESH. We also demonstrated the efficacy and robustness of our optimization methodology by producing 3D SWNoC architectures that can perform equally or better than the fully connected 3D MESH with significantly less number of vertical links. For the case of 50% reduction in vertical links, the optimized 3D SWNoC achieves 25% lower EDP compared to fully vertically connected 3D MESH NoC. Finally, the robustness fo SWNoC is analyzed by comparing against other regular and irrelgular 3D NoCs.
4 COUNTERACTING ELECTROMIGRATION OF TSV-BASED 3D NOC VIA SPARE-VERTICAL LINK (SVL) ALLOCATION

4.1 Overview of Electromigration Challenge for TSV-enabled 3D System

The performance improvement of 3D NoCs can be attributed to the presence of vertical links (generally TSVs) between the planar dies. As a result, the electrical characteristics of the TSVs have significant effects on the overall NoC performance. The RC characteristics of the TSVs change over time and increases nonlinearly depending on the traffic load [26]. Under heavy traffic and high link utilization, the increase in resistance (R) of any TSV is more prominent than the change in capacitance [126] [62]. The net effect of heavy TSV utilization is that it increases the delay of that particular TSV. In the worst case, the delay of the TSV can even increase more than one clock cycle period. This excessive delay is equivalent to ‘TSV failure’.

Figure 4.1 (left) Workload induced stress and electromigration phenomenon for the TSVs. The through current causes stress and forces the TSV material to undergo electromigration at the landing pad and creates void. (right) TSV resistance profile with time.
The ‘wear-out’ effect of TSVs carrying heavy traffic load decreases the overall lifetime of the chip. The average length of time before a TSV fails is referred to as the mean-time-to-failure (MTTF). The effect of high TSV utilization can be incorporated as the increase in TSV resistance over time [26]–

\[
R(t) - R_0 = A \ln \left( \frac{t}{t_0} \right)
\] (4.1)

where \( R_0 \) is the initial TSV resistance and mainly depends on the fabrication process and component materials. Similarly, due to process variation, cracks and voids are created in the TSVs [127] [68]. The parameter, \( t_0 \), is the time when the size of void becomes equal or greater than to the cross-sectional area of the TSV. The value of aging coefficient, \( A \), depends on the amount of current that passes through TSVs in unit time. In this work, \( A \), was taken to be 0.05 kΩ/log(s) based on the analysis reported in prior work [26] [128]. The parameter \( R(t) \) is the value of the resistance at time \( t \) in (4.1) and any increment in \( R(t) \) will give rise to TSV delay. When the resistance parameter, \( R(t) \), increases to a value such that the delay of the TSV crosses the allowable limit, then that value of \( t \) is estimated to be the MTTF of that particular TSV. (Note that we estimate the MTTF in this manner because a more accurate computation of MTTF requires a statistically significant sample size of TSVs and actual measurement data.) The absolute value of this allowable delay depends on the performance requirement of the application under consideration. The parameter, \( R(t) \), is also related to the traffic density and hence the TSV utilization. The TSV utilization can be expressed as utilized cycles/simulated cycles and by multiplying with the parameter \( t \), we can calculate the traffic-induced TSV utilization. Essentially, this captures the amount of stress that is applied to the TSVs due to the traffic.
We considered 2:1, 4:1 and 8:1 TSV serialization/deserialization ratio and found that the 4:1 configuration achieves the best trade-off between timing and area overhead. Hence, in this work, we considered a 4:1 TSV serialization/deserialization ratio. For the TSMC 28 nm technology node, we calculated the delay of the TSV including the serialization/deserialization overhead and the worst-case crosstalk noise from neighboring TSVs, to be 141 ps. As mentioned above, $R(t)$ increases over time depending on the workload, which in turn increases the worst-case delay of the TSV. In this work, without loss of generality, we set the allowable timing budget to be an additional 20% of the initial TSV delay. Once TSV delay passes this limit, we assume that the TSV has failed. We consider the time to reach this 10% additional delay as the MTTF of that particular TSV [63] [26].

![Figure 4.2 Effect of non-homogeneous TSV utilization for 3D SWNoC with CANNEAL benchmark. (a) TSV utilization (in percentage of cycles that TSV actively transmit signals to total simulated cycles). (b) MTTFs for individual TSVs.](image)
4.1.1 TSV Model to Study Workload-Induced Stress

To study the effects of workload-induced stress and crosstalk noise among the TSV-enabled links, we model the TSV as a lumped RC $\pi$-network [129]. Figure 4.3(b) shows the complete model. The parameters $R_{TSV}$ and $C_T$ indicate the TSV resistance, and self-capacitance respectively. The resistance corresponding to the leakage current of the TSV is represented by $R_s$. The values of respective TSV parameters are calculated using the model from [129]. The effects of workload-induced electromigration of the TSV material is modeled by introducing additional resistance, $REM$, in series with the TSV resistance. The parameter, $R_{EM}$, is zero at $t = 0$, and then depending on the workload characteristics, starts to increase.

![Figure 4.3](image.png)

Figure 4.3 (a) TSV-based vertical link placement between routers from adjacent tiers. TSVs are placed in bundle e.g. 3x3, 4x4 etc. configuration. (b) TSV model to study the electromigration effect.

4.1.2 Counteracting Electromigration of TSV via Spare TSV Allocation

To overcome such performance penalties arising from TSV failure, researchers have investigated spare TSV (sTSV) allocation and sharing scheme for 3D ICs [23] [70]. The initial sharing algorithm was developed based on the idea of utilizing one extra TSV for each of the functional TSVs [70]. However, the reliability improvement comes at the expense of double TSV count and significant area overhead.

Different TSV sharing schemes, proposed in the literature (as described in Chapter 2, Section 2.3), improve the performance of 3D ICs and hence, the overall reliability as well. However, the
allocation of spare TSVs for 3D NoCs need to consider additional constraints arising from the physical NoC design perspective. In a 3D NoC, TSVs are placed in a bundle to enable a single vertical link (VL). Depending on the physical placements of routers and cores of 3D NoCs, these VLs maintain considerable physical distance between them. Hence, sharing TSVs among these VLs is not feasible due to the physical design and timing constraints. In addition, if one TSV fails in a VL, then the achievable performance of the whole link is affected, which in turn degrades the overall NoC performance. Hence, we focus on spare-vertical link (sVL) allocation instead of individual spare TSVs.

In this work, to analyze the reliability issues of 3D NoC, we evaluate the performance of a 3D NoC with workload-induced VL failure. We perform a comparative reliability analysis study of different 3D NoCs with VL failure. Then we describe the sVL allocation mechanism for a 3D NoC. We formulate sVL allocation as an optimization problem to minimize the performance penalty due to TSV-based VL failure. At the same time, we focus on improving the lifetime of the overall 3D NoC. We demonstrate two different algorithms viz. greedy and exhaustive search to allocate the sVLs in a 3D SWNoC to compensate for the performance penalty due to VL failure. We also compare the performance of both algorithms in terms of quality of the solution and computation.
time. We show that based on the domain knowledge of a 3D NoC, we can develop computationally efficient algorithms whose performances are similar to exhaustive search, a naïve approach.

4.2 Problem Formulation for Spare VL Allocation Problem

Given a set of $m$ functional VLs $F$ and budget size of spare-VLs $n$ ($n > 0$, $n \ll m$), we want to select the subset of $n$ functional VLs out of $m$ those when provided with one spare-VL each will maximize the reliability (lifetime) of the 3D NoC. We can experimentally compute the quality of a given sVL allocation solution by running a simulation. This is an instance of a combinatorial optimization problem with an unknown cost function, where the quality of a given solution can be computed only by making a simulator call. Here, the term ‘solution’ refers to a particular 3D NoC configuration incorporated with spare-VLs for $n$ functional VLs.
4.2.1 Computational Challenges

The main challenge here is that we have a huge number of possible solutions or NoC configurations \( \binom{m}{n} \) to allocate spare-VLs among the functional links. A naive approach is to enumerate all possible solutions; compute the quality of each solution via simulator call; and pick the best solution. However, the simulator call is expensive in terms of both time and memory requirements. Hence, this exhaustive search approach to quantify the performance and lifetime of each of the candidate NoC configuration is infeasible for practical purposes.

<table>
<thead>
<tr>
<th>Algorithm 4.1: Greedy Spare-VL Allocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: \textbf{Input:} ( F = \text{set of } m \text{ functional VLs, } n = \text{budget for spare-VLs,} )</td>
</tr>
<tr>
<td>2: \textbf{Output:} ( S ), the best set of ( n ) functional VLs that gets spares</td>
</tr>
<tr>
<td>3: \textbf{Initialization:} initialize solution set ( S = \emptyset )</td>
</tr>
<tr>
<td>4: \textbf{for} each greedy step ( = 1 ) to ( n )</td>
</tr>
<tr>
<td>5: \quad \textbf{for} each choice ( x \in F )</td>
</tr>
<tr>
<td>6: \quad \quad \text{value} ( (x) = \text{simulator_call} \ (S \cup x) )</td>
</tr>
<tr>
<td>7: \quad \textbf{end for}</td>
</tr>
<tr>
<td>8: \quad x^* = \text{arg max}_{x \in F} \text{value} \ (x)</td>
</tr>
<tr>
<td>9: \quad S = S \cup x^* \quad \text{// Functional VL } x^* \text{ gets spare}</td>
</tr>
<tr>
<td>10: \quad F = F \setminus x^* \quad \text{// } x^* \text{ is removed from } F</td>
</tr>
<tr>
<td>11: \textbf{end for}</td>
</tr>
<tr>
<td>12: \textbf{return} ( S )</td>
</tr>
</tbody>
</table>

\*\text{simulator\_call} is a procedure that calculates and returns the network performance and lifetime for a given NoC configuration, benchmark suite, and routing algorithm through extensive experiments.

4.2.2 State Space Search Formulation

We solve the sVL allocation problem using a state-space search formulation, where the simulations guide the search process. Each state in our search space is a particular NoC configuration allocated with spare-VLs and consists of a set \( S \subseteq F \), where \( S \) is a partial or complete solution. Our search space is a 3-tuple \( \langle I, A, T \rangle \), where \( I \) is the initial state function that returns the initial search state
$S = \emptyset$ meaning solution set is empty; $A$ is a finite set of actions (or search operators) corresponding to growing the partial solution $S$ by one element from $F \setminus S$; and $T$ is the terminal state predicate that maps search nodes to $\{1, 0\}$ indicating whether the node is a terminal or not. Each terminal state in the search space corresponds to a complete solution ($|S| = n$, where $|S|$ denotes the total number of candidates of $S$), while non-terminal states correspond to a partial solution ($|S| < n$). Thus, the decision process for constructing a complete solution corresponds to selecting a sequence of actions leading from the initial state (none of the spare-VLs are allocated) to a terminal state (all the $n$ spare-VLs are allocated). In principle, we can employ any heuristic search procedure (e.g., greedy, beam search) guided by simulations.

### 4.2.3 Greedy Search for Spare-VL Allocation

This is the simplest search procedure. We start with an empty solution set $S$. In each greedy step, we add the spare-VL from $F \setminus S$ to the solution set $S$ that when provided with a spare link, it improves the reliability by maximum amount. We repeat this greedy selection step until $S$ is a complete solution ($|S| = n$). The time complexity of greedy search is $O(m*n - n^2)$ simulator calls. Algorithm 2 provides the pseudo-code for greedy spare allocation.

The greedy search is able to produce highly effective sVL allocation that can significantly improve the reliability of the 3D NoC. This effect of greedy sVL allocation was observed through experimental studies as the cost function is unknown and we need to find solutions via simulator calls. The allocation policy to allocate a spare (if spare-VL budget allows) to the first functional VL that fails with a given functional and spare VL-based 3D NoC configuration is highly effective. Intuitively, if we don't allocate spare to the functional VL that is expected to fail first, it will result in a cascade of VL failures reducing the lifetime of the chip drastically.
4.2.4 Domain Knowledge for Sound Pruning

In 3D NoC enabled many-core chips, some VLs experience heavy traffic and high utilization as the underlying routing algorithm tries to find shortest paths between source and destination cores via these links. As a result, those VLs with high utilization undergo heavier stress, and introduce additional delay in the path and fail more quickly when compared to others. Moreover, this is not an independent phenomenon: one VL failure can increase the time to failure of a neighboring VL leading to a clustering effect as workload of the neighboring links increase. For example, in Fig 3, we show the traffic densities and the MTTF values of all the VLs for a 64-core and four-layer 3D SWNoC for the CANNEAL benchmark (one of the PARSEC benchmark with highest traffic injection load and skewed traffic). We can see that the traffic densities of VLs 17 to 32 (we call this region as critical region) are significantly higher than that of the others and expectedly, their mean-time-to-failure (MTTF) values are significantly lower.

Our key insight is that for a small budget size \( n \) (say less than the number of critical VLs), the spares should be allocated to some of the critical VLs only and there is no benefit for allocating spares to non-critical VLs (chip will fail due the failure of all critical VLs). We can use this domain knowledge to "soundly" prune the search space of possible solutions for the spare-VL allocation problem. Let \( H \subseteq F \) correspond to the critical VLs and the total number of critical VLs is \( h \) where \( h = |H| \). If we consider complete solutions from \( H \) only (i.e., subsets of size \( n \) from \( H \)), we can still retain the optimal solution. In other words, we get huge computational savings without losing any accuracy due to sound pruning. For exhaustive search, we can consider \( \binom{h}{n} \) instead of \( \binom{m}{n} \) candidate solutions, where \( h < m \). For greedy search, we only consider the VLs from \( H \) for spare allocation (number of actions \( |A| \) will be relatively smaller).
For the rest of the experiments and analysis, we denote the baseline greedy and exhaustive search by Greedy-Full and Exhaustive-Full respectively. In addition, these techniques enabled with domain knowledge-based pruning are named as Greedy-Restricted and Exhaustive-Restricted.

### 4.3 Performance Evaluation of 3D NoCs with sVL Allocation

In this section, present a detailed reliability analysis in the presence of spare-VL insertion. To characterize the performance of spare VL allocation algorithm, we consider the runtime of the algorithm and the lifetime of 3D SWNoC as the evaluation metric. First, we characterize the performance of Greedy and Exhaustive search algorithms for sVL allocation. Next, we analyze how the sVL allocation improves the EDP profile and hence, the reliability and lifetime of 3D NoCs.

#### 4.3.1 Experimental Setup

To evaluate the performance of different NoCs, we use a cycle-accurate NoC simulator that can simulate any regular or irregular 3D architecture [103] [7]. Our system consists of 64 cores and 64 network routers equally partitioned in four layers. The length of each packet is 64 flits and each flit consist of 32 bits. The routers are synthesized from an RTL level design using TSMC 65-nm CMOS process in Synopsys™ Design Vision. All router ports have a buffer depth of two flits and each router port has four virtual channels in case of irregular NoC. The NoC simulator uses wormhole routing, where the data flits follow the header flits once the router establishes a path. For regular 3D mesh-based NoC, XYZ-dimension order-based routing is used [120]. For irregular architectures such as the SW network, the topology-agnostic Adaptive Layered Shortest Path
Routing (ALASH) algorithm is adopted [119]. The energy consumption of the network routers, inclusive of the routing strategies, was obtained from the synthesized netlist by running Synopsys™ Prime Power, while the energy dissipated by wireline links was obtained through HSPICE simulations, taking into consideration the length of the wireline links. We consider four SPLASH-2 benchmarks, namely, FFT, RADIX, LU, and WATER [122], and five PARSEC benchmarks, namely, DEDUP, VIPS, FLUIDANIMATE, CANNEAL, and BODYTRACK (BT) [123] in this performance evaluation. These benchmarks vary in characteristics from computation intensive to communication intensive in nature and thus are of particular interest in this work.

To quantify the effects of sVL allocation, we first define the lifetime of the 3D SWNoC. Whenever any functional VL fails in a 3D SWNoC, the average hop count increases and hence, the network latency and EDP increase as well. Eventually, the EDP of the 3D SWNoC may be higher than fault free 3D MESH, where it can no longer be considered as an efficient architecture. At this point, the 3D SWNoC loses its architectural advantages over a 3D MESH. We consider the time required to reach this configuration as the lifetime of the 3D SWNoC.

4.3.2 Lifetime of 3D SWNoC

In an NoC, the (latency, energy, EDP per message) and reliability are interrelated. Without any VL failure, the 3D SWNoC, optimized for achieving high performance, initially (at t = 0) shows lower EDP values compared to a standard 3D MESH architecture [16]. However, due to VL failure, the network configuration changes and consequently, the network latency and energy consumption increase progressively. At a certain time, the EDP of 3D SWNoC increases beyond that of a fault-free 3D MESH. At this point, the 3D SWNoC can no longer be considered more efficient than a fault-free 3D MESH (without any link failure), and the corresponding time is termed as the lifetime.
of the 3D SWNoC. The failure of VLs reduces the lifetime of 3D NoC significantly. To express the definition of 3D SWNoC lifetime formally-

\[
\text{lifetime}_{3D\text{-SWNoC}} = \left\{ t : (EDP_{3D\text{-SWNoC}}(t)) = EDP_{3D\text{-MESH}}(t=0) \right\}
\]  

(4.2)

Hence, the term, lifetime of 3D SWNoC, indicates the period of time up to which it operates with a lower EDP than a conventional 3D MESH. In this context, the EDP of 3D MESH is considered merely as a reference point for comparative study. However, any other definition of lifetime of 3D SWNoC (e.g. 10% increase in EDP value), is applicable as well.

The lifetime of 3D NoC unifies both the performance (EDP) and reliability of the NoC at the same time. When the SWNoC reaches its lifetime, its performance can no longer be considered as efficient as the fault-free case.

Figure 4.5 lifetime determination algorithm is explained with normalized EDP profile of 3D SWNoC w/ and w/o sVL allocation. As an example, lifetime calculation for 3D SWNoC with DEDUP benchmark has been plotted. The EDP of 3D MESH-0 sVL (dotted line) corresponds to time t=0 and extended only for reference purpose.

(Figure 4.5) illustrates the lifetime computation procedure for a 3D SWNoC incorporated with 8 sVLs for the DEDUP benchmark. This particular configuration is chosen as an example; however, the procedure is applicable for any other 3D NoC and benchmark. For the reference purposes, the
EDP profile of the original 3D SWNoC (without any sVL) is also plotted. The EDP of 3D SWNoC is normalized with respect to the EDP value of fault free 3D MESH. To help illustrate the lifetime computation procedure more clearly, a dotted horizontal line is drawn in (Figure 4.5), which we call the lifetime line. This line corresponds to 100% EDP value for the fault free 3D MESH (at t=0).

We have calculated the lifetimes of 3D SWNoC and 3D SWNoC with 8 sVLs as marked with respective vertical lines. The lifetimes of these NoCs are the projection of the intersection point of the lifetime line and the corresponding EDP profile lines of 3D SWNoC with and without spare-VLs on the time axis. These are marked as L3D SW and L3D SW with 8-sVL in the figure. The lifetime of other 3D NoCs can be calculated in a similar way.

### 4.3.3 Greedy vs. Exhaustive Search for sVL Allocation

For the sVL allocation, we explored two different allocation algorithms as explained in section IV: Greedy search and Exhaustive search. The allocation algorithms are named as Greedy-Full and Exhaustive-Full (The suffix ‘Full’ is added to differentiate the algorithms with domain knowledge-based pruning, which we will introduce later).

![Figure 4.6 Lifetime of the 3D SWNoC as a function of the number of sVL for the CANNEAL benchmark: Greedy-Full vs. Exhaustive-Full.](image-url)
Figure 4.6 plots the lifetime of the 3D SWNoC with sVL allocation using Greedy-Full and Exhaustive-Full algorithms for different number of spare-VLs for the CANNEAL benchmark. From the figure, we can see that both Greedy-Full and Exhaustive-Full sVL allocation algorithms achieve the same lifetime for the 3D SWNoC. Note that, greedy search (as expected) takes significantly less computation time to produce the solution when compared to exhaustive search.

To understand the reason for this behavior, we first need to understand the details of the sVL allocation procedure. To be more specific, the VL failure sequence and its effects on NoC performance need to be explored.

If any functional VL fails, then the workload of this particular VL negatively affects the other neighboring VLs and as a result, the EDP increases rapidly. Consequently, allocation of sVLs to the functional VL, which fails first, is expected to minimize the NoC performance penalty. If sVL is allocated without following the VL failure sequence, then the allocation effect may not be visible on both the EDP profile and lifetime at all.

To explain this behavior in more detail, we consider the case of the CANNEAL benchmark for the 3D SWNoC and number the 48 VLs serially starting from 1 to 48 for a 64-core system (as shown in Fig. 3(Figure 4.2)). For 8 spare-VLs, the sVL allocation solution from exhaustive search corresponds to assigning spares to functional VLs numbered 26, 22, 27, 10, 42, 43, 7, and 6. Somewhat surprisingly, greedy search also produced the same sVL allocation solution. Our experimental analysis showed that the greedy search produces sVL allocation solutions that can significantly improve the reliability of the 3D NoC. The allocation policy to allocate a spare (if the sVL budget allows) to the first functional VL that fails with a given functional and spare VL-based 3D NoC configuration is highly effective. Intuitively, if we do not allocate spare to the functional
VL that is expected to fail first, we will be faced with a cascade of VL failures, which will reduce the lifetime of the chip drastically. For example, the VL failure sequence without any spares allocated is 26, 22, 27, 10, 32, 30, 25, 18 and so on. Greedy search allocates the first spare to functional VL 26. The VL failure sequence after assigning spare to VL 26 is 22, 26, 27, 23, 32, 30, 31, 25, 18 and so on. Greedy search allocates the second spare to functional VL 22. Continuing this policy, greedy search assigns spares to the same set of functional VLs as done by the exhaustive search. We found this behavior to be consistent across all the benchmarks.

4.3.4 Domain Knowledge for Pruning the Search Space

The time to compute the sVL allocation solution grows as the number of functional VLs ($m$) and the number of sVLs ($n$) increases for both exhaustive search and greedy search. The time complexities of exhaustive search and greedy search (in terms of the number of simulator calls) are $O\left(\binom{m}{n}\right)$ and $O(mn - n^2)$, respectively. For example, for a 64-core 3D NoC with $m=48$ and $n=8$, the total solution exploration times for exhaustive and greedy search are 377,348,994$q$ and 356$q$ respectively (here $q$ corresponds to the computation time of a single simulator call which is currently ~7 min in the current experimental setup using a machine configured with Intel Core i7-4700MQ processor and 8 GB RAM running at a clock frequency of 2.4 GHz). Therefore, our sVL allocation algorithms may not scale for large-scale 3D NoC. We consider using domain knowledge of the workload of different functional VLs to prune the solution space as described in Section V. In 3D NoC, the workload of some functional VLs (say critical VLs) is much higher than the others and as a result their failure probabilities are higher too. Intuitively, when the sVLs budget ($n$) is small, it is beneficial to allocate spares to some of the critical VLs only because the chip will fail due to a cascade of critical VL failures.
We select a subset of critical VLs (say $H$) out of the $m$ functional VLs that we will consider for allocating spares and prune the remaining ones. Pruning can improve the computational efficiency of solving the sVL allocation problem, however, may potentially compromise the accuracy of solutions depending on the amount of pruning. We can consider varying amounts of pruning from $|H|=n$ (only one candidate solution) to $|H|=m$ (no pruning) to trade-off speed and accuracy of producing sVL allocation solutions. A simple pruning strategy to achieve this goal is as follows: rank all the functional VLs according to their workload; select the top-$|H|$ VLs to be considered for spare allocation; prune the remaining $m-|H|$ VLs. We can use both exhaustive search and greedy search to find the solution from this restricted set of candidate solutions. We refer to the exhaustive and Greedy sVL allocation algorithms as Exhaustive-Restricted and Greedy-Restricted, respectively.

Figure 4.2 shows the traffic densities of all the VLs for a 64-core 3D SWNoC consisting of four planar layers for the CANNEAL benchmark. It can be noted that the traffic densities of some VLs (critical VLs) are significantly higher than that of the others. To identify the critical VLs, we rank VLs according to workload and sort the highest workload ones. In this particular work, we consider
16 critical VLs. This number is chosen considering the worst-case VL failure scenario where all 16 critical VLs are placed in between two adjacent planar dies and if all of them fail together, then the NoC becomes completely unrouteable. Therefore, we prune all the non-critical VLs, a total of 32 out of 48 (other than 16 critical VLs). In other words, \(|\mathcal{H}|=16\) corresponding to 16 high workloads carrying VLs, which is significantly smaller compared to \(m=48\) (total number of VLs). We found that with this setting, the search algorithms with pruning (Exhaustive-Restricted and Greedy-Restricted) produce the same sVL allocation solutions as their counterparts without any pruning (Exhaustive-Full and Greedy-Full) for different number of spare-VLs \(n=1\) to any number of upper limit). In other words, we do not lose accuracy due to pruning. We do not show these results for the sake of brevity. The main benefit of pruning is that it will improve the computational efficiency of producing sVL allocation solutions using Exhaustive and Greedy search. As an example, Figure 4.7(a) shows the runtime comparison of Greedy-Full and Greedy-Restricted for the different number of sVLs. Similarly, Figure 4.7(b) shows the runtime comparison of Exhaustive-Full and Exhaustive-Restricted. We can see that the computational gains are significant due to pruning, but without losing any accuracy.
4.3.5 Effects of Spare-VL Allocation on 3D NoC

Whenever a spare-VL is allocated to a functional VL, the sVL carries the traffic when the corresponding functional VL fails. This minimizes the effect of VL failure on 3D NoC performance degradation and essentially helps in maintaining lower EDP value over longer period of time. However, there exists an upper limit for the sVL number, beyond which the advantages of sVL allocation can no longer be pronounced. We call this number as the optimum number of spare-VLs.

![Normalized EDP profile for 3D SWNoC with different number of spare VLs allocation for the (a) CANNEAL, (b) DEDUP, (c) VIPS benchmark.](image)
Depending on the benchmark and NoC configuration, the optimum number of sVL varies. In this work, we consider 3D SWNoC architecture as the testbed for evaluating the performance of sVL allocation. However, subsequent experiments and analysis are equally applicable for other 3D NoC architectures as well.

4.3.6 Optimum Number of spare VLS

In this section, we evaluate the effects of different number of sVLs on the 3D NoC performance. Figure 4.8(a), (b) and (c) demonstrate the normalized EDP of 3D SWNoC with time for CANNEAL, DEDUP, and VIPS benchmarks respectively. Similar to the previous experiments, we have considered these three benchmarks as the representative of high, medium, and low injection benchmarks from the PARSEC and SPLASH-2 suites. All the EDP values are normalized with respect to the EDP of fault free 3D MESH with no sVLs allocated to it at $t=0$.

From these figures, we can see that the EDP remains unchanged up to a certain point and after that, it increases when the functional VLs start failing. This happens due to the fact that initially no functional VL fails and EDP remains constant up to a certain time. Subsequently, VLs from the critical region (as defined in section V.E) having high traffic density start failing. In such a link failure scenario, the traffic of the failed VLs is carried by the neighboring VLs along with their own traffic. This has two kinds of negative effects. Firstly, the EDP and the network latency of the NoC increases due to a critical link failure. Secondly, the neighboring functional VLs also fail quickly which further degrades the NoC performance. As a result, the EDP increases at a faster rate.

Another interesting result is that as the number of allocated spare-VLs increases, the EDP profile shifts towards the right on the time scale. This implies that the 3D SWNoC with sVL allocation
can maintain a particular EDP level for a longer period of time. Expectedly, the lifetime of 3D SWNoC also increases with sVL allocation. In addition, we can see that the difference between the EDP profiles on the time axis decreases gradually as the sVL number increases. For the CANNEAL benchmark, the right-most EDP is found to be for 8 sVLs. We can see that even if we increase the number of sVLs beyond 8 for CANNEAL, the EDP profile doesn’t shift to the right anymore. This implies that any further improvement of EDP profile is not possible and we call this scenario as the saturation effect of sVL allocation. Similarly, for 3D SWNoC with DEDUP and VIPS benchmarks, the EDP profile gets saturated for 14 sVLs.

4.3.7 Saturation of Lifetime Improvement

In this work, we have considered one-to-one correspondence between sVLs and functional-VLs where any sVL replaces one functional VL regardless of the workload intensity. Allocation of such sVLs increases the traffic carrying capability of the critical VLs and improves the lifetime of the 3D NoC. As an example, Figure 4.9 plots the percentage of lifetime improvement of 3D SWNoC for the CANNEAL benchmark with different number of sVLs allocation. Note that similar lifetime improvements are observed for other benchmarks as well.

From the figure, we can see that as the number of allocated sVLs increases, the lifetime of the 3D SWNoC also increases. Initially, the gain of lifetime is almost linear with the number of allocated sVLs and later, the gain increment decreases and improvement saturates after some point. Allocation of sVL increases the combined lifetime of the particular VL (consists of sVL and
functional VL in this case), which helps to minimize the network latency and EDP degradation due to VL failure.

In general, most critical VLs fail early when compared to the other VLs. If the sVLs are allocated to the critical VLs, then they help in significantly increasing the lifetime of the NoC. However, the lifetime gain saturates as the number of allocated sVL crosses a certain number. This happens due to the fact that the combined lifetime of some critical VLs even with the sVL allocation is shorter than other non-critical VLs. Consequently, even if we allocate sVLs to these non-critical VLs, they do not improve the EDP beyond what is achieved already. We may see benefits if we consider a more general sVL allocation problem, where we consider allocating more than one spare-VL to a functional VL. We defer this for future work. It is important to note that similar effects are also observed for DEDUP and VIPS benchmarks as well (in these cases, the saturation effect was observed for 14 sVLs). However, we have omitted plotting such repetitive results and analysis.

![Figure 4.9 Effect of sVL allocation on 3D SWNoC for the CANNEAL benchmark. The improvement of lifetime of 3D SWNoC initially increases linearly and saturates beyond 8-sVL allocation. The gain is normalized w.r.t. the initial lifetime of 3D SWNoC at t=0.](image)

Figure 4.9 Effect of sVL allocation on 3D SWNoC for the CANNEAL benchmark. The improvement of lifetime of 3D SWNoC initially increases linearly and saturates beyond 8-sVL allocation. The gain is normalized w.r.t. the initial lifetime of 3D SWNoC at t=0.
4.4 Summary of Spare Vertical Link Allocation for 3D NoC

In this work, we explored a robust NoC design methodology to counteract the electromigration-induced failure of TSVs considering the nonhomogeneous workload distribution of a 3D NoC. We also proposed a spare-VL allocation mechanism to address the performance degradation and lifetime shortening problem due to VL failure. We showed that with a small number of spare-VLs, we could exploit NoC domain knowledge to develop efficient and computationally inexpensive algorithms to explore optimal solution. The proposed spare-VL allocation significantly improve the reliability and lifetime of the 3D NoC.
5 JOINT EFFECTS OF ELECTROMIGRATION AND CROSSTALK NOISE ON TSV-ENABLED 3D NOCS

5.1 Overview of Electromigration and Crosstalk Noise for 3D NoC

The anticipated performance gain of a TSV-enabled 3D NoC can be compromised due to TSV failure [71] [17]. In the previous chapter, we have seen that workload distribution among the vertical links (VLs) in a 3D NoC varies widely and imposes significant challenges for reliability. The resultant stress causes electromigration effects for TSVs and thereby increasing their resistances, which in turn increase the delay [26] [128]. Ultimately, it will negatively affect the NoC performance.

In addition to electromigration, TSVs in a 3D NoC are placed in bundles to create the VLs [130] [131]. In each VL, the cross-coupling effects from the neighbors increase the worst-case delay of each TSV [132]. The combined effects of workload-induced stress and crosstalk lead to significant performance and reliability concerns for TSV-enabled 3D manycore chips. The failure of TSVs changes network configuration, increases the network latency, degrades system performance, and ultimately reduces the overall NoC lifetime.

To improve the reliability of any 3D IC, spare TSV allocation has been advocated in several works [47] [23]. However, the joint effects of workload-induced stress and crosstalk were not considered for spare TSV allocation. In this work, our goal is to explore the combined effects of these two TSV reliability issues on the performance and robustness of a 3D NoC. Consequently, we evaluate
the efficacy of a spare TSV allocation scheme by considering the joint effects of workload-induced stress and crosstalk to enhance the lifetime of the 3D NoC.

5.2 TSV Reliability From 3D NoC Perspective

In this section, we discuss the TSV failure mechanism from a 3D NoC perspective by considering the joint effects of workload-induced stress and crosstalk noise on the TSV lifetime. As the workload-induced stress has already been focused on previous chapter, we mainly focus on the crosstalk noise here. Finally, the joint effects are explored.

5.2.1 Effects of Workload-induced Stress

As discussed in previous section, in a 3D NoC, the vertical links (VLs) consisting of TSVs undergo large variation in workload distribution depending on the NoC architecture and application. Nonhomogeneous workload distribution also results large variation in their lifetime termed as mean-time-to-failure (MTTF). To consider the joint effects of workload induced stress and crosstalk noise, we consider the outcome from the previous section for workload induced stress on a 3D NoC. We incorporate the similar TSV lifetime (MTTF) definition for this work e.g. the time corresponds to 10% increase in delay of TSV-enabled links for is considered as their lifetime [26].

Figure 5.1 Different kinds of TSV placement configuration. Most popular approach is the grid-based placement (e.g. 3×3, 3×4, 4×5, 3×5 and so on).
5.2.2 TSV Placement Configuration

Figure 5.1 shows different kinds of TSV placement configurations used in 3D manycore chips. Depending on the placement configuration, the crosstalk noise among the neighbor TSVs vary significantly. Intuitively, row-based TSV placement results lowest amount of effective crosstalk noise, however, it also consumes large area associated with the TSV pitch. Among them grid-based TSV placement is the most popular [25]. In our analysis, we consider this configuration as the baseline to explore the effects of the crosstalk noise on TSV lifetime.

5.2.3 Effects of Crosstalk Capacitance

Crosstalk noise in a TSV bundle mainly depends on two important factors, viz., the exact location of the TSV (which determines the proximity to other TSVs), and the bit patterns. For a TSV based 3D IC, this causes variation in delay. To illustrate this, we consider the most popular TSV placement strategy viz. the grid-based arrangement. Figure 5.2 shows the TSV placement for a 3x3 configuration and the associated worst-case crosstalk capacitances for each TSV in the bundle.

For a single signal transition between a pair of adjacent TSVs, the crosstalk capacitance is assumed to be $C_c$. As seen from the figure, the center TSV is subject to the highest crosstalk noise (10 $C_c$), while the adjacent neighbors face the second highest (7 $C_c$) and finally, the corner TSVs has the minimum crosstalk (4.5 $C_c$). Depending on the amount of worst-case crosstalk any TSV

**Figure 5.2 Placement of TSVs in a grid. (a) The worst-case crosstalk capacitance for the center TSV (TSV 5). (b) The aggregation of worst case $C_c$ values for each TSV.**

78
experiences, TSVs can be categorized in three classes – Type 1 (worst case crosstalk experience is $10 \ C_c$), Type 2 (worst case crosstalk experience is $7 \ C_c$) and Type 3 (worst case crosstalk experience is $4.5 \ C_c$). Figure 5.2(c) shows the different categories of TSVs.

While the self-capacitance is same for all TSVs in the bundle, the crosstalk depends on the bit pattern and the position of the TSV in the bundle. Hence, the delays of the TSVs in a bundle also vary. We have considered a 3x3 bundle to visually demonstrate the crosstalk effects, however, all other grid-based configurations i.e., 3x4, 4x4, 4x5, 5x5 etc. are supersets of this particular placement strategy. Hence, the worst-case crosstalk as found in this 3x3 grid pattern is applicable to the other configurations also.

5.2.4 Joint Effects of Workload and Crosstalk on TSV MTTF

Due to the above-mentioned variation in the TSV resistance arising out of workload-induced stress and the crosstalk, the delay of the TSV also increases. At a certain point of time, the delay increases beyond the acceptable limit imposed by the timing constraint of the system. This scenario can be considered as a failure of TSV, and the corresponding time is termed as the mean-time-to-failure (MTTF). In general, 10% increase in delay is considered as the failure of a TSV [26] [63]. In this work, we also consider the time corresponding to this 10% increase in delay as the MTTF for any TSV. The overall MTTFs depend on the workload-induced stress as well as the crosstalk capacitance. If we do not consider any of these factors, the MTTFs show large deviation from the actual values.
5.3 Problem Formulation

In this work, we consider the small-world network enabled 3D NoC (3D SWNoC) as the suitable architecture for exploring the effects of electromigration and crosstalk on the overall reliability and lifetime. This particular architecture is chosen due to its better performance and robustness compared to other existing counterparts [20]. Our target is to explore how the workload variation and crosstalk affect the overall performance and reliability of a 3D SWNoC. Note that, the subsequent analysis presented in this work is general, and it can be equally applied to any other TSV-enabled 3D NoC.

5.3.1 Reliability Metric: Lifetime of 3D NoC

Without any VL failure, the 3D SWNoC initially (at $t = 0$) exhibits lower energy-delay-product (EDP) values compared to a standard 3D MESH [20]. However, due to VL failure, the EDP increases progressively and at a certain time, the EDP of 3D SWNoC increases beyond that of a fault-free 3D MESH. At this point, the 3D SWNoC is no longer more efficient than the 3D MESH, and the corresponding time is termed as the lifetime of the 3D SWNoC. To express it formally:

$$\text{lifetime}_{3D \text{ SWNoC}} = \left\{ t : (\text{EDP}_{3D \text{ SWNoC}} (at \ t=t)) = \text{EDP}_{3D \text{ MESH}} (at \ t=0) \right\}$$  \hspace{1cm} (5.1)

Hence, the term, lifetime of 3D SWNoC, indicates the period of time up to which it operates with a lower EDP than a conventional 3D MESH. In this context, the EDP of 3D MESH is considered merely as a reference point for comparative study. However, any other definition of lifetime of 3D NoC (e.g. 10% increase in EDP value), is applicable as well.
5.3.2 Problem statement

In this work, we consider a 3D NoC with multiple planar dies. The inter die communication take place though $N$ number of VLs (each VL having $n$ fTSVs). The 3D NoC executes a set of applications and due to the nature of the applications, the workload distribution of the VLs is nonhomogeneous and some of the VLs experience higher traffic load than the others. Moreover, individual TSVs in each bundle are affected by crosstalk capacitance. Consequently, the MTTFs of these VLs vary depending on the position of individual TSVs in a bundle and the workload-induced stress of the VLs. Traditionally, TSV failure in a 3D IC has been addressed by spare TSV allocation. In this work, we consider the joint effect of workload-induced stress and TSV crosstalk capacitance on a spare TSV allocation methodology to improve the overall MTTF of a 3D NoC.

In most of the existing sTSV allocation algorithms TSV failures arising out of clustering faults, fabrication and bonding defects, uniform-random faults and timing faults are explored [133]. In contrast, we target the TSV failure due to the nonhomogeneous workload-induced stress that degrades NoC performance over time along with the crosstalk capacitance.

5.3.3 TSV Fault Model with Workload Variation and Crosstalk

To determine the delay of any TSV, we consider the circuit model shown in Figure 5.3 [134] [129], where $R_{\text{workload}}$ is the net increase in the resistance value, $R(t)$, due to workload-induced stress. In the figure, the parameters- $R_{TSV}$, $L_{TSV}$, $C_{TSV}$ indicate the TSV resistance, inductance, and self-capacitance respectively, while $C_c'$ denotes the effective crosstalk of a TSV. The parameter, $R_{\text{workload}}$, has value of zero at $t = 0$, and then depending on the workload characteristics, starts to increase. Intuitively, the TSV with the worst-case crosstalk in the highest utilized VL is expected to have the maximum delay and the lowest MTTF. Consequently, this particular combination
becomes the main bottleneck for the 3D SWNoC from both the performance and reliability perspectives.

![Figure 5.3 Equivalent circuit model of TSV to determine the effects of workload-induced stress and crosstalk.](image)

5.4 Spare TSV Allocation for Reliability Improvement

The goal of sTSV allocation is to replace the failed fTSVs with spare ones so that a certain level of achievable performance is maintained. However, the spare allocation methodology should incorporate the effects of both electromigration (workload-induced stress) and crosstalk. To undertake a detailed analysis regarding this joint effect, any existing sTSV allocation mechanism can be adopted. In this work, we consider the sTSV allocation algorithm proposed in [135], where the main goal was to handle EM-induced faults. For this algorithm, at each step, the TSV with the minimum MTTF was explored, and subsequently allocated with the spares until the upper limit on the budget is reached or the target system MTTF is achieved. However, to employ this algorithm in our work, we make two main modifications. First, we consider the joint effects from workload-induced stress and crosstalk capacitances together on the TSV MTTFs. Secondly, our goal is to maximize the lifetime of the 3D NoC by considering the EDP profile and TSV MTTFs. In order to identify the TSV with the lowest MTTF, at any particular time, \( t \), we consider the utilization
pattern $u_k(t)$, and corresponding crosstalk capacitance value of that particular TSV. Allocating a spare to the TSV with lowest MTTF ensures the minimal effect of TSV failures on the NoC EDP profile and thereby the lifetime of 3D NoC is maximized.

5.5 Experimental Results and Analysis

In this section, we analyze the performance of the 3D SWNoC in the presence of workload-induced stress and crosstalk noise. Subsequently, we discuss the efficacy of a sTSV allocation mechanism in the presence of these two factors to enhance the lifetime of the 3D SWNoC.

5.5.1 Experimental Setup

We consider a Chip Multiprocessor (CMP) consisting of 64 cores and 64 routers equally partitioned into four layers. In each die, 16 cores (and associated routers) are placed at regular intervals in a grid pattern and TSVs act as VLs in between layers. The TSV dimensions for this work are- length of 15 μm, diameter of 2.3 μm, and pitch of 30 μm [26] [25]. The TSV model parameters i.e. $R_{TSV}$, $L_{TSV}$, $C_{TSV}$, $C_c$ are calculated following [136] [137]. The associated logic circuitry is synthesized from an RTL level design using the ST 28-nm standard cell library in Synopsys™ Design Vision.

To evaluate the performance of 3D NoCs, we use a cycle-accurate NoC simulator that can simulate any regular or irregular 3D architecture. The length of each message is 64 flits and each flit consist of 32 bits. The NoC simulator uses wormhole routing, where the data flits follow the header flits once the routers establish a path. For the 3D SWNoC network, the topology-agnostic Adaptive Layered Shortest Path Routing (ALASH) algorithm is adopted [119]. The energy consumption of the network routers was obtained from the synthesized netlist by running Synopsys™ Prime Power,
while the energy dissipated by wireline links was obtained through HSPICE simulations. We consider CANNEAL, DEDUP and VIPS benchmarks from PARSEC [122], and FFT, RADIX and WATER from SPLASH-2 suite in this performance evaluation [123]. These benchmarks vary in characteristics from being computation-intensive to communication-intensive in nature and thus are of particular interest in this work.

### 5.5.2 Analyzing TSV MTTF

In this section, we analyze the effects of crosstalk capacitance and workload on TSV delay and MTTF.

#### 5.5.2.1 Determination of Effective $C_c'$ Values

To evaluate the effects of neighboring TSVs on $C_c'$ value, we consider in total six different TSV placement configurations i.e. 1x8, 2x4, 2x5, 3x3, 3x4, 3x5 and apply all the possible bit patterns.

Table 5.1: Worst case and effective crosstalk capacitances for TSV bundles with different configurations

<table>
<thead>
<tr>
<th>Grid Patterns</th>
<th>Worst $C_c'$</th>
<th>Effective $C_c'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1x8</td>
<td>$4C_c$</td>
<td>$1.56C_c$</td>
</tr>
<tr>
<td>2x4</td>
<td>$7C_c$</td>
<td>$2.25C_c$</td>
</tr>
<tr>
<td>3x3</td>
<td>$10C_c$</td>
<td>$2.46C_c$</td>
</tr>
<tr>
<td>3x4</td>
<td>$10C_c$</td>
<td>$2.63C_c$</td>
</tr>
<tr>
<td>3x5</td>
<td>$10C_c$</td>
<td>$2.81C_c$</td>
</tr>
</tbody>
</table>

*In this work, $C_c$ is the amount of crosstalk for a single signal transition between two neighboring TSVs.*

Table 5.2: Effective crosstalk capacitance for different types of TSVs in a 3x3 grid pattern-based placement

<table>
<thead>
<tr>
<th>Types of TSVs</th>
<th>TSVs</th>
<th>Worst $C_c'$</th>
<th>Effective $C_c'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>$10C_c$</td>
<td>$4.04C_c$</td>
</tr>
<tr>
<td>2</td>
<td>2, 4, 6, 8</td>
<td>$7C_c$</td>
<td>$2.74C_c$</td>
</tr>
<tr>
<td>3</td>
<td>1, 3, 7, 9</td>
<td>$4.5C_c$</td>
<td>$1.77C_c$</td>
</tr>
</tbody>
</table>
(signals) to determine the effective \( C_{c'} \) values. Figure 5.4 shows the variation of \( C_{c'} \) for these configurations (inset shows their respective envelopes). The vertical axis indicates the percentage of occurrence for any particular \( C_{c'} \) value.

As seen from the figure, the grid pattern 1x8 has the lowest value for \( C_{c'} \) (clustered towards the left) while 3x5 shows the maximum possible \( C_{c'} \) value (having higher \( C_c \) values more frequently) among all of them. In a 1x8 configuration, the worst-case crosstalk has a value of 4 \( C_c \) while the grid patterns of 3x3, 3x4, 3x5 have the worst value of 10 \( C_c \). In addition, the latter grid patterns have 1, 2, and 3 candidates respectively those can potentially face the highest 10 \( C_c \) values. As a result, the \( C_{c'} \) distribution curve for 3x5 shifts to the right on the horizontal axis. For other placement configurations e.g. 4x4, 4x5, 5x5, 5x6, 6x6 etc., the worst-case \( C_{c'} \) remains same as 3x3 configuration (10 \( C_c \)), however the number of TSVs having the highest \( C_{c'} \) values increase. As a result, the effective \( C_{c'} \) also increases for those grid patterns as well. In this work, we consider a 32-bit datalink (flit width is 32 bits) and 4:1 serialization ratio for the TSVs. Hence, we choose the 3x3 grid pattern as it satisfies the VL design requirements for further MTTF analysis.

From the practical point of view, the TSVs in a 3D NoC do not always encounter the worst-case crosstalk noise. Hence, we also calculate the effective \( C_{c'} \) value by considering all possible bit

![Figure 5.4 Variation of crosstalk capacitances for different TSV grid-based placement configurations](image-url)
patterns for the TSVs in a bundle. Table 5.1 shows the effective values for \( C_{e}' \) for the above mentioned TSV bundles. As expected, the 3x5 grid has the highest amount of average effective \( C_{e}' \) value among all the configurations while the lowest value is observed for 1x8 grid pattern. However, due to the comparatively longer inter-TSV wire-length requirement, the this 1x8 configuration is not considered in practice.

### 5.5.2.2 Joint Effects of Crosstalk and Workload on MTTFs

The joint effects of workload-induced stress and crosstalk capacitance can result in the large variation in TSV MTTFs for the 3D SWNoC. To explain this, we consider the 3x3 TSV grid pattern from Figure 5.2. For this grid pattern, all nine TSVs are numbered from 1 through 9 as shown in this figure. These TSVs can be categorized into three types based on their worst-case crosstalk capacitances (similar classes can be identified for 3x4, 3x5, 4x4, 5x5 placements as well). The first type (Type 1) consists of the center TSV of each bundle (TSV 5, shown in Figure 5.2), which has the highest amount of \( C_{e}' \) value (shown in Table 2). The second (Type 2) and third (Type 3) categories are the adjacent neighbors of Type 1 TSVs (TSVs 2, 4, 6, 8), and the corner TSVs of the bundle (TSVs 1, 3, 7, 9) respectively. These classes i.e. Type 1, 2, and 3 have in total 1, 4 and 4 candidates respectively for this particular 3x3 grid pattern. Table 5.2 shows the worst case and effective (averaged over all possible bit patterns for the 3x3 grid) crosstalk capacitances. We consider the effective capacitance values for the following MTTF distribution analysis.

Now, in addition to the crosstalk capacitance, the resistance of the TSVs also increases due to the workload-induced stress and resulting electromigration (equations (4.1) and (4.2) in Section 4.1). These two together increase the TSV delay and influence the overall MTTF. Figure 5.5 shows the variation of MTTF vs. the occurrence frequency (in percentage) for the above-mentioned three
types of TSVs considering the joint effects of workload and crosstalk capacitance. We consider all
the TSVs belonging to different VLs. The horizontal axis indicates the normalized MTTFs
(normalized with respect to the lowest value of the system). For this MTTF analysis purpose, we
have considered three benchmarks from the PARSEC suite viz. CANNEAL, DEDUP, VIPS.
These benchmarks are chosen as the representative of high, medium, and low traffic-injection
benchmarks.

From these figures, it is seen that the MTTFs vary widely among different types of TSVs.
Expectedly, Type 1 has the maximum crosstalk capacitance values, and consequently, the lowest
MTTFs, while type 3 TSVs have the minimum effective \( C_c' \) and the resultant highest MTTFs.

Figure 5.5 Variation of MTTF for three types of TSVs in 3x3 grid configuration for (a) CANNEAL,
(b) DEDUP, and (c) VIPS benchmark.
Workload variation and resultant nonuniform stress result in different resistance values for the TSVs belonging to various VLs. On top of this, the presence of crosstalk capacitance, each TSV within the VL possesses different MTTF. Hence, large variation in MTTFs is observed for different TSVs across the VLs.

5.5.3 TSV MTTFs with only Workload-induced Stress

From the above analysis, it is clear that the MTTFs are significantly affected by the crosstalk capacitance. However, if we only consider the workload-induced stress without considering the effects of crosstalk capacitance from the neighboring TSVs, then the resultant TSV MTTFs may have significant deviation from the actual one. To explain this, Figure 5.6 plots the TSV MTTF variation for the CANNEAL benchmark considering both these cases. Other benchmarks also have similar distributions; however, we have omitted presenting those for brevity. The MTTFs with only considering workload-induced stress are marked as $EM_{\text{only}}$ while the joint effects of workload and crosstalk are considered for the $EM+C_c$. From the figure it is clear that the $EM_{\text{only}}$ overestimates the MTTFs (clustered towards right in the figure) and fails to capture the actual MTTF distribution. Actually, when we consider the joint effects of EM and crosstalk then the MTTF distribution shifts towards left in Figure 5.6. Hence, incorporation of crosstalk capacitances
with workload variation is required for comprehensive analysis of TSV MTTFs and overall reliability.

5.5.3.1 Reliability of 3D NoC with Spare TSV Allocation

The wide variation in TSV MTTF ensues the performance degradation of 3D SWNoC during the execution of any application. In this subsection, the joint effects of workload variation and crosstalk are discussed for allocating the sTSVs to enhance the reliability of 3D SWNoC.

5.5.3.2 Joint Effects of EM and Cc on Spare TSV Allocation

The joint effects from workload-induced stress and crosstalk noise on TSV MTTF also affect the performance of sTSV allocation methodology. To explain this, we consider two cases for the sTSV allocation algorithm outlined in Section 5.4. For the first case, during the allocation, we only consider the effect of workload-induced stress without considering crosstalk, while for the second, the joint effects are considered. The average lifetime improvements of 3D SWNoC with these two cases considering the PARSEC and SPLASH-2 benchmarks mentioned above are plotted in Figure 5.7. The improvement in lifetime is normalized with respect to the lifetime of 3D SWNoC without any spare. The two lifetimes are marked as $sTSV_{EM}$ and $sTSV_{EM,C}$ respectively.

![Figure 5.7 Improvement of lifetime of 3D SWNoC with the spare TSV allocation algorithm by considering only workload-induced stress ($sTSV_{EM}$) and the joint effects ($sTSV_{EM,C}$).](image)
It is evident that $sTSV_{EM,C_C}$ shows significant lifetime improvement over $sTSV_{EM}$ for any number of spares. Both the $sTSV_{EM}$ and $sTSV_{EM,C_C}$ algorithms explore the TSV with the lowest MTTF and allocate spare to minimize the negative effect of TSV failure. However, $sTSV_{EM}$ only considers workload-induced stress without evaluating any effects from crosstalk capacitance. When we only consider the EM effect arising due to the workload-induced stress then the individual TSVs in a VL are indistinguishable from each other (resistance of each TSV is affected equally by the workload). Consequently, the spare allocation mechanism assigns sTSVs to the whole bundle. This fails to exploit the full benefits of spares as some of the TSVs in the bundle may not require spares (whose MTTFs are comparatively higher, viz. Type 2 and Type 3 TSVs from Figure 5.5). Instead of allocating spares to Type 2 and Type 3 TSVs in a highly utilized VL, the spare allocation should assign spares to Type 1 TSV of the next highly utilized VL. In contrast, for $sTSV_{EM,C_C}$ allocation, the exact TSV within the VL having the lowest MTTF is identified by exploiting the knowledge of crosstalk capacitance and workload-induced stress. This ensures spare allocation to the most critical TSVs belonging to various VLs rather than allocating spares to all the TSVs in a specific bundle. This strategy ensures the best possible usage of spares and thereby minimizing the adverse effect of TSV failure on the whole NoC. Hence, the $sTSV_{EM,C_C}$ performs better than $sTSV_{EM}$ for any spare budget and maximizes the NoC lifetime.

### 5.5.4 Lifetime Improvement of TSV based 3D NoC

In this subsection, we analyze how the lifetime defined in (5.1) of the 3D SWNoC varies with different applications and given spare budgets. Figure 5.8 shows the normalized maximum lifetime achieved by varying the number of sTSVs considering the earlier-mentioned SPLASH-2 and
PARSEC benchmarks for the 3D SWNoC. The lifetime is normalized with respect to that value of the SWNoC without any sTSV allocation.

From the figure it is seen that as the number of sTSVs increases, the lifetime improvement also increases. For example, with sixteen sTSV, the maximum lifetime improvements achieved for these six benchmarks vary from 43% to 72%. The highest and lowest improvements were observed for WATER and VIPS benchmarks respectively. These improvements are achieved with the sTSV\_EM\_Cc spare allocation algorithm (from previous section) considering the joint effects of crosstalk noise and workload-induced stresses. If we consider a 64-core system with 48 VLs and 8 TSVs for each VL (using a 4:1 mux for enabling 32-bit fit width), then a budget of 16 sTSVs represents ~5% of the overall TSVs (total 48*8 TSVs). In addition, from Figure 5.8, it is seen that this 5% spares can achieve on average 59% lifetime improvement for TSV enabled 3D SWNoC. However, if the joint effects of crosstalk noises and workload-induced stresses are not considered then the lifetime improvement can be reduced significantly (as seen from Figure 5.7).

5.6 Summary of Reliability Analysis for TSV-enabled 3D NoC

Reliability is a major concern for TSV based 3D NoCs due to nonhomogeneous workload-induced stress in the vertical links. In the presence of crosstalk effects from neighboring TSVs in a bundle,
the MTTF of each TSV and the overall lifetime of the NoC degrade further. In this work, we have addressed the combined effects of electromigration arising due to workload-induced stress and crosstalk capacitance on TSV MTTF and hence the lifetime of the 3D NoC. Without considering the crosstalk capacitance, the TSV MTTF and NoC lifetime can be overestimated by at least one order of magnitude. We have also demonstrated that by allowing only a 5% spare TSV budget, the incorporated spare TSV allocation methodology can result in ~59% lifetime improvement of a 3D SWNoC for SPALSH-2 and PARSEC benchmarks considered in this work.
6 PERFORMANCE-RELIABILITY TRADE-OFFS FOR 3D TSV-ENABLED MANYCORE CHIP

6.1 Overview of Reliability Analysis for TSV-enabled 3D NoC

The anticipated performance gain of any 3D NoC-enabled manycore chip is compromised due to the potential failures of TSV based vertical links (VLs) (a VL corresponds to a bundle of TSVs) [23] [24]. The non-homogeneous workload-induced stress among the TSV-enabled vertical links (VLs) in 3D NoCs causes large variations in the mean-time-to-failure (MTTF) distribution. Such a failure of TSV-based VLs leads to performance degradation over time [61] [29]. One possible solution to enhance the lifetime of these VLs is to incorporate a suitable power management strategy to reduce the workload-induced stress, which will also improve the overall reliability of the 3D NoC.

To improve the energy-efficiency of manycore systems, voltage-frequency island (VFI)-based power management techniques have shown promise in the past [138] [139] [140]. In addition, the average voltage and current levels, and the temperature profile of the system are also reduced by incorporating VFI enabled power management [77] [141] [142]. The reduction in operational temperature reduces the thermal hotspots in chip, and as a result, significantly improves the overall reliability. The benefits of incorporating VFI-based power management methodology are threefold: (i) The reduction of V/F levels (and hence the amount of current passing through the VLs) will reduce the stress level for TSVs; (ii) The power-management strategy is expected to reduce energy consumption for the 3D manycore chip over its entire lifetime; (iii) The reduced
temperature profile enabled by the power management will significantly increase the MTTF values of the TSVs.

Alternatively, adaptive routing methodologies (marked as Adaptive later in this work) have been proposed for improving the lifetime of TSVs for a 3D manycore system [143] [144] [29]. However, adaptive routing mainly focuses on redistribution of the traffic to reduce the workload-induced stress of most highly utilized VLs. During adaptive routing, some of the routing paths can be longer relative to the shortest paths, and as a result, latency of the system increases. Hence, the enhancement in lifetime enabled by adaptive routing mainly comes from redistribution of traffic at the expense of higher performance penalty.

The main idea of spare allocation is to place additional TSVs along with functional TSVs (f-TSVs) in a bundle so that if f-TSVs fail, then s-TSVs can replace them. Hence, allocation of spares enhances the reliability of the 3D systems and maintains the same level of system performance for longer duration of time [71]. This s-TSV allocation mechanism is applicable to 3D NoCs as well.

However, among all the previously mentioned reliability improvement strategies, only the VFI-based power management reduces the voltage levels of VLs dynamically depending on the workload. Hence, we can conjecture that the power management is potentially more effective in enhancing the lifetime of 3D NoC-based manycore chips when compared to adaptive routing and s-TSV allocation approaches.

To summarize, we make three main contributions in this work:

- Considering the NoC design challenges and associated concerns, we present the reliability analysis of TSV-enabled 3D NoC. In order to do so, we consider both the workload-induced stress and cross-talk noise from neighboring TSVs in a bundle. We show that if we only consider the
workload-induced stress for analyzing the 3D NoC reliability, we would be targeting only one part of the overall problem. The combined effects of the electromigration arising out of workload-induced stress and the crosstalk capacitance need to be considered to determine the realistic lifetime of a 3D NoC.

- We propose to enhance the lifetime of TSV-based VLs in 3D NoCs by incorporating VFI-based power management, and thereby improving the lifetime of the whole system. The proposed reliability improvement strategy is independent of the specific 3D NoC architecture and application type.

- We discuss TSV failures due to non-homogeneous workload-induced stress from the 3D NoC perspective. We quantify how the MTTF (mean to time to failure) values of the TSV-enabled VLs are affected by different types of applications, and how this variation in MTTFs is translated to determine the lifetime of a given 3D system.

- We perform comprehensive experiments to evaluate and analyze the effectiveness of VFI-based power management when compared to adaptive routing and s-TSV allocation methods to enhance the reliability of 3D SWNoC for SPLASH-2 and PARSEC benchmark suites.

Figure 6.1 Illustration of the reliability-performance trade-off analysis for TSV-enabled 3D NoCs.
Figure 6.1 shows the workflow diagram to evaluate the reliability and performance trade-offs of 3D SWNoC. We consider a 3D SWNoC and perform performance evaluation, and thermal profiling by incorporating VFI-based power management and adaptive routing methodologies. Next, the MTTF distribution for the VLs are determined considering the TSV utilization, workload-induced stress, and the operating temperature of the system. Finally, the reliability and performance trade-offs incorporating VFI-based power management and adaptive routing are compared against spare TSV allocation methodology.

6.2 TSV Reliability

In this section, we briefly discuss TSV failure scenarios and the MTTF variation with non-homogeneous workload across the VLs, and their effect on the performance of the 3D SWNoC.

6.2.1 Failure of TSVs under Workload-Induced Stress

In the presence of stress induced by high current, a TSV undergoes electromigration, and as a result, the resistance of the TSV increases [63] [128]. This causes a significant increase in the delay of the TSV, which may eventually compromise the timing specification of the design [145] [146].

To explain this, in Figure 6.2(a), we plot the workload distribution for all the TSV-enabled VLs in a 3D SWNoC for one of the well-known PARSEC benchmarks, viz., DEDUP. To characterize the individual workload for each VL, we define the active VL utilization factor, $v_n(t)$, at any given instant of time, $t$, for the $n^{th}$ VL as –

$$v_n(t) = \frac{Busy(t,n)}{Cycles(t,n)}, \quad \forall \ n \in \{1,2,...,N\}$$  \hspace{1cm} (6.1)

where, $N$ is the total number of VLs; $Busy(t,n)$ is the number cycles during which $n^{th}$ VL has actively transferred data; and $Cycles(t,n)$ indicates the total number of simulation cycles. From
the Figure 6.2(a), it can be seen that the workload of the individual VLs (and hence, for individual TSVs) vary widely.

Such variation of workloads across the VLs causes non-homogeneous stress in them. From the physical perspective, the workload-induced stress causes the TSV material (in general Cu) to undergo electromigration at the junction of the TSVs and their respective landing pads. In addition, higher workload makes the migration effects more pronounced. As a result, a void is created at the TSV landing pad resulting in a reduction of the effective TSV cross-sectional area and forcing the current to pass through the TSV barrier material (e.g., TiN). The net outcome of the increase of TSV utilization is that, the resistance and consequently, the delays of the TSVs also increase. At a certain point of time, the delay increases beyond the acceptable limit imposed by the timing constraint of the system. This scenario can be considered as the failure of a TSV, and the corresponding time is termed as the MTTF. In general, a 10% increase in resistance (or delay) can be considered as the failure of a TSV [63] [128].

Figure 6.2 Workload variation and its effects on the vertical links (VLs) of a 64-core 3D SWNoC for the DEDUP benchmark. (a) Time average VL utilization metric ($v_{avg}$) for all the VLs numbered from 1 to 48. (b) The MTTF distribution for the same system. All the MTTF values are normalized with respect to the lowest MTTF value in the system. The maximum MTTF can be as high as 6.2 times of the lowest MTTF.
6.2.2 Effects of TSV Utilization and Current Density on MTTF

Due to the workload-induced stress, the TSV resistance remains almost constant up to a certain point of time, and then starts increasing in a nonlinear manner [63] [128] [26]. The resistance of TSVs under the electromigration effect can be modeled as -

\[ R(t) - R_0 = A \ln\left(\frac{t}{t_0}\right), \quad t > t_0 \]  \hspace{1cm} (6.2)

where, \( A = \frac{\rho_B}{4\pi t_B} \) and \( t_0 = \frac{t_{cu}\pi r_{TSV}^2}{aF} \). \hspace{1cm} (6.3)

In the above equations, the parameters \( R(t) \) and \( R_0 \) refer to the resistance values at times \( t = t \) and \( t = 0 \), respectively. The parameter \( A \) is called the aging coefficient, and \( t_0 \) is the time when the void (created at the TSV and landing pad junction) becomes greater than the TSV cross sectional area. The other parameters, namely \( \rho_B, t_{cu}, t_B, \) and \( r_{TSV} \), denote the TSV barrier resistivity, copper thickness, barrier thickness, and radius of the TSV, respectively. The parameter \( aF \) denotes the portion of the vacancy flow, which leads to electromigration and generates the void under the TSV. While the aging coefficient, \( A \), is independent of the stress condition, the parameter \( t_0 \) depends on the stress condition. More specifically, the vacancy flow parameter \( aF \) depends on the amount of current flowing through TSVs and can be expressed as - \( aF \propto i^n \), where \( i \) is the net amount of through current, and the exponent \( n \) has value \( 2.0 \pm 0.2 \) from the experimental results verified using Black’s equation [147] [26]. It is to be noted that the time parameter, \( t \), in the above equations denotes the active utilization time of the TSVs. Hence, the MTTFs of the TSV-enabled VLs directly depend on the active VL utilization metric (\( v_n(t) \)). Any variation in \( v_n(t) \) results in wide differences in the MTTF across the VLs in the system.

To analyze the effects of active VL utilization metric, \( v_n(t) \) on MTTF of the VLs more elaborately, in Figure 6.2(b), we plot the MTTF distribution considering the TSV utilization pattern from
Figure 6.2(a) (for the same 3D SWNoC architecture with the DEDUP benchmark). For comparative study, the MTTF values are normalized with respect to the lowest MTTF value in the system. From the figure, it can be seen that for one single benchmark, the MTTF of one particular VL can be as high as 6.2 times of the lowest MTTF. In such a scenario, the TSV-enabled VLs with lower MTTF values will fail relatively earlier (the lower ones fail first) than others and become the main bottleneck for the whole system. Additionally, the largest value of \( v_n(t) \) results in the lowest MTTF for the VLs and vice versa. Hence, reduction in active utilization of a TSV results in a slower increase in its resistance, enhanced MTTF values, and thereby leading to improved reliability of the overall system.

In addition, from Equations (6.2) and (6.3), we can observe that the increase in resistance is a linear function of \( t_0 \). However, the parameter, \( t_0 \), itself depends on the net amount of current \( (i) \) passing through the TSV cross-sectional area and is related to the amount of current by the equation- \( t_0 \propto i^n \) (where \( n \) has value 2.0±0.2). Hence, reduction of the net amount of current \( (i) \) can reduce the overall electromigration effect for the TSVs.

### 6.2.3 Effects of Temperature on the MTTF of TSVs

The electromigration in TSVs also depends on the operational temperature. Any increase in temperature results in enhanced vacancy flow of the TSV material (denoted by \( \alpha F \) in equations (2) and (3)), accelerating the void growth at the TSV landing pad. This will eventually reduce the parametric value of \( t_0 \), which indicates the time when the resistance starts to increase non-linearly. This temperature dependence of the parameter, \( t_0 \), has been modeled as [26]-

\[
t_0 = \frac{t_{Cu} \pi r_{TSV}^2 k_B T}{\alpha N_i D_{eff} f_0 Z^* \rho_{Cu}} i^n \exp \left( \frac{E_A}{k_B T} \right)
\]  

(6.4)
where the parameters- \( D_{eff0}, N_i, Z^*, E_A, k_B, \) and \( T \) are the diffusivity coefficient, intrinsic carrier concentration of the carriers, total number of effective carriers, activation energy, the Boltzman constant, and operating temperature, respectively. The other parameters (i.e., \( t_{Cu}, r_{TSV}, \alpha, \rho_{Cu} \)) are the same as in the previous section. The most interesting part of this dependency is that, the parametric value of \( t_0 \) depends exponentially on the effective temperature value of the TSVs. Hence, reduction of the operational temperature of the TSVs will result in a significant increase in their MTTF values.

### 6.3 MTTF Improvement

Based on the above discussion, we can infer that to maximize the lifetime of TSV-based VLs, we can follow several approaches to reduce the electromigration effect. First, to reduce the stress of individual TSVs (and VLs), the active utilization metric, \( v_n(t) \), can be reduced especially for the highly utilized VLs. Second, the operating voltage and hence, the energy consumption, can be minimized to reduce the net amount of current \( (i) \) passing though the TSVs. As a result, the MTTF value of any TSV is expected to increase. Most importantly, if the operating temperature of the system can be reduced, then significant improvement in MTTF will be achieved. Hence, in this work, to improve the TSV MTTF, we aim at incorporating different design techniques to reduce the TSV utilization metric \( (v_n(t)) \), voltage/current levels \( (V/F \) levels), and overall system temperature \( (T) \).

### 6.4 Problem Setup

In this section, we consider the small-world (SW) network-based 3D NoC architecture as a testbed NoC architecture and subsequently describe the problem statement.
6.4.1 Problem Formulation

In this work our target is to compare and contrast the VFI-based power management and adaptive routing as enabling techniques to enhance the lifetime of 3D NoCs. We also compare the performance of these methods with the previously introduced spare TSV allocation mechanism.

**VFI-based Power Management:** Our goal is to improve the reliability (lifetime) of a 3D NoC-enabled manycore chip by reducing the workload-induced stress for highly utilized VLs via a power-management mechanism. Specifically, we consider a VFI-based power management approach to achieve this goal. Given a VFI-enabled 3D NoC architecture, an application that determines the workload characteristics, and allowed performance penalty (say p% execution time penalty), the goal of the VFI control mechanism is to allocate appropriate V/F for each of the VFI s such that the overall energy consumption is minimized subject to the performance penalty constraint. The reduction of energy consumption by incorporating VFI-based power management improves the MTTF values for the VLs of the 3D NoC. In addition, the reduction in energy consumption reduces the temperature, which in turn helps in improving the MTTF. Our target is to establish the performance-reliability tradeoffs for the VFI-enabled 3D SWNoC.

**Adaptive Routing Alternative:** The nonhomogeneous traffic density for the TSV-based VLs creates large variation in the MTTF distribution (as seen from Figure 6.2(a) and Figure 6.2(b)). Some of these VLs have really large TSV utilization metric, \( v_n(t) \), and consequently, very low MTTF values. Hence, the traffic carried by the VLs with very low MTTF needs to be decreased. However, as the total amount of traffic for a particular application cannot be reduced, reducing the amount of traffic for a particular VL will increase the workload for others. As a result, the MTTF of those other VLs may decrease. For the 3D NoC architectures with multiple planar layers, the
traffic density in the middle layers is generally higher than in the top and bottom ones. Hence, the 
VLS between these layers experience much higher traffic density. As a result, the MTTF of these 
VLS are significantly lower, and ultimately, they influence the lifetime of the whole system. Our 
target is to redistribute the traffic load in such a way that the lower MTTF values in the critical 
region will increase. We may have to sacrifice the VLS with higher MTTFs to some extent. 
However, the increase in MTTF for the critical VLS is more significant than the reduction in 
MTTFs of other VLS. Hence, the average MTTF of the whole system will still improve. 
Redistribution of the traffic can be incorporated in the routing algorithm. We modify the routing 
algorithm to enhance the MTTF of highly utilized VLS in order to improve the lifetime of the 
whole system [29].

6.4.2 Lifetime as a Reliability Metric

To evaluate and quantify the overall reliability of the manycore system, we consider lifetime as 
the relevant metric. In this section, we explain how we measure the lifetime of a 3D NoC.

Without any VL failure, the 3D SWNoC initially (at $t = 0$) performs better in terms of energy-
delay-product (EDP) compared to a standard 3D MESH. As TSV based VLS fail in a 3D SWNoC 
due to any one of the above-mentioned phenomena, the EDP increases progressively and at a 
particular time, the EDP of SWNoC increases beyond that of the fault free 3D MESH. At this point, 
the 3D SWNoC can no longer be considered as an efficient architecture, and the corresponding 
time is considered as the lifetime of 3D SWNoC. Hence, the lifetime of a 3D SWNoC for any 
application can be defined as the time when its performance (EDP) matches to that of a 3D MESH 
with no link failure, given that both NoCs utilize equal amount of resources (i.e., total number of
horizontal and vertical links, TSVs, planar dies, cores, routers, and overall footprint is also same).

More formally-

\[
\text{lifetime}_{3D\ NoC} = \{ t : (EDP_{3D\ NoC\ (at\ t=t)} = EDP_{3D\ MESH\ (at\ t=0)}) \} 
\] (6.5)

Following this definition, the lifetime of 3D SWNoC (or any other 3D NoC that performs better than 3D MESH) indicates how long the 3D NoC performs better than a 3D MESH with VL failure. Hence, this lifetime definition can be considered as the reliability metric for 3D NoCs. It is to be noted that we consider the 3D MESH EDP value as the reference for determining the lifetime of 3D SWNoC for a comparative performance study. However, any other lifetime definition can be adopted for exploring similar reliability and performance trade-offs.

### 6.5 Reliability Improvement Methodologies

To improve the reliability of the 3D SWNoC enabled manycore chips, we explore two different methodologies. In this section, we describe the details of the *VFI-based power management* methodology, and then the *Adaptive routing* strategy. In addition, the s-TSV allocation scheme has also been outlined to address the workload-induced faults of TSVs.

#### 6.5.1 Dynamic Voltage-Frequency Islanding (DVFI) Methodology

In VFI-based power management methodology, the main idea is to cluster groups of cores along with the associated routers and links depending on the computation and communication patterns. Cores and network elements with similar workloads and traffic patterns can share the same Voltage/Frequency (V/F) values, without introducing significant performance penalties. We incorporate the DVFI strategy from [148] for the 3D SWNoC. It is to be noted that the main focus of this work is not to introduce any new methodology for VFI clustering or VFI-control policy,
but to incorporate and analyze the effects of VFI based power management for reliability improvement of 3D NoCs. Hence, any other suitable DVFI methodology can be incorporated to achieve the main goal of this work.

6.5.1.1 VFI Clustering Methodology

In any VFI-based system, the primary goal is to cluster cores with similar behavior so that all the cores and the associated network elements (routers and links) in a particular VFI can benefit from coordinated V/F tuning. In this respect, instructions per cycle (IPC) per core and traffic statistics are employed to capture the utilization and communication behavior of the cores respectively. In a traditional VFI-based system, the average IPC and traffic statistics were found to be effective in creating VFI clusters [148]. However, in the presence of volatile computation and traffic patterns, the VFI clusters should be created according to the time-varying IPC and traffic statistics. Prior work has explored several VFI clustering methodologies [148]. In this work, we create VFI clusters using the hybrid clustering algorithm to minimize each VFI’s intra-cluster variation in the time-varying IPC and traffic statistics following the work of [148]. However, any other clustering mechanism can be adopted for VFI creation. In-depth analysis of clustering algorithms is beyond the scope of this paper.

6.5.1.2 Feedback based VFI Controller

In traditional single core/router DVFS heuristics, the CPU utilization and the inter-tile communication are considered as the two main application characteristics for determining appropriate V/F levels [148]. These techniques were applied to DVFI by using the average VFI information in conjunction with offsets that take into account the variance of this information
In this work, we adopt this DVFI control policy and briefly outline the salient features of this DVFI control mechanism.

In order to determine a suitable V/F pair for all the cores and routers within a VFI, we obtain a metric that incorporates information from all elements in the VFI. We start by defining the core utilization of core $i$ ($u_i(t)$) and link utilization for the link between cores $k$ and $l$ ($lu_{kl}(t)$):

$$u_i(t) = \frac{Busy(t,i)}{Cycles(t,VFI_j)}, i \in VFI_j$$  \hspace{1cm} (6.6)

$$lu_{kl}(t) = \frac{Flits(t,k,l)}{Cycle(t,VFI_j)}, l \in VFI_j$$  \hspace{1cm} (6.7)

where $VFI_j$ is the set of cores in VFI cluster $j$, $Busy(t,i)$ is the number of core busy cycles for window $t$ for core $i$, $Cycles(t,VFI_j)$ is the number of total cycles for window $t$ for VFI $j$, and $Flits(t,k,l)$ is the number of flits received by core $l$ from core $k$ during window $t$. Each VFI $j$ has its own V/F controller that operates independently and calculates a metric, $m(t)$, using the core and link data; this metric is used in the V/F determination based on the information during each window $t$:

$$m(t) = \omega_u \sum_{v_i \in VFI_j} \frac{u_i(t)}{|VFI_j|} + \omega_c \sum_{v_k \notin VFI_j} \frac{lu_{kl}(t)}{InterLinks(VFI_j)}$$  \hspace{1cm} (6.8)

where $|VFI_j|$ is the number of cores in VFI $j$, $InterLinks(VFI_j)$ is the number of inter-VFI links connected to VFI $j$, and $\omega_u$ and $\omega_c$ are the weights for the utilization and communication parts respectively. Intuitively, $m(t)$ is the weighted sum of the VFI’s average core utilization and average incoming inter-VFI link utilization. The weights $\omega_u$ and $\omega_c$ are calculated as the proportion of cores utilization to traffic during the current time window $t$. 
Based on the value of $m(t)$, a thresholding mechanism is used to calculate the predicted V/F for the next time window, $t + 1$, and the V/F is adjusted for the VFI accordingly. The following thresholding mechanism is used:

$$m(t) = \begin{cases} 
VF_0, & \text{if } m(t) + \Delta M > R_1 \\
VF_1, & \text{if } R_1 \geq m(t) + \Delta M > R_2 \\
\vdots & \vdots \\
VF_{K-1}, & \text{if } R_{K-1} \geq m(t) + \Delta M > R_K \\
VF_K, & \text{if } R_K \geq m(t) + \Delta M 
\end{cases} \quad (6.9)$$

where $\Delta M$ is a threshold offset that adjusts the mapping of the metric to a particular V/F by a fixed value, $VF_n$ represents the $n^{th}$ V/F value (with $VF_0$ being the nominal V/F), and $R_n$ represents the ratio between the $n^{th}$ V/F value and the nominal V/F.

To summarize the relationship between $\Delta M$ and $m(t)$, increasing $\Delta M$ gives rise to an increase in $m(t)$ resulting in higher V/F values. Also, decreasing $\Delta M$ lowers $m(t)$, which results in lower V/F values. Therefore, the higher values of $\Delta M$ would typically lead to lower execution times and higher energy dissipation and vice versa. We use $\Delta M$ to compensate for tracking errors and variability between cores within the VFI. We reiterate that the details of the VFI controller implementations and performance analysis is beyond the scope of this work. A comprehensive analysis can be found in [148].

### 6.5.1.3 Reliability Improvements Offered by DVFI

In general, by incorporating VFI, the effective voltage and frequency levels for cores, network elements, and associated VLs are reduced for the VFI-enabled 3D SWNoC. As a result, the overall EDP profile improves. The reduced voltage levels result in lower current, which minimizes the electromigration of the VLs. The lifetime of the TSVs are inversely proportional to their respective
current levels ($MTTF$ values $\alpha_i n$, where $n=2 \pm 0.2$). Hence, the reduction of system EDP and improvement in TSV MTTF results in higher lifetime for VFI-enabled 3D system.

In addition, DVFI reduces core power, network power and hence, overall system power leading to significant reduction in the system temperature. Such lowering of temperature can improve the MTTF values of the TSVs significantly. Hence, the MTTF values of the TSV based VLs increase, and the lifetime of the chip improves.

The effective average voltage-frequency level, overall system EDP, and the operating temperature depend on the types of applications running on the system and the overall system configuration. Hence, the net amount of improvement in lifetime also varies with types of applications, and the underlying NoC architecture. The adopted SWNoC architecture is more energy efficient (and show lower temperature profile) and hence, is expected to improve the lifetime of the VFI-enabled system significantly. Figure 6.3 summarizes the reliability improvement process for DVFI enabled SWNoC architecture.

![Figure 6.3 Illustration of DVFI implementation methodology for 3D SWNoC, and the establishment of performance-reliability trade-off. Incorporating DVFI strategy improves both EDP profile and the overall reliability (lifetime) by improving the VL MTTF values.](image)
6.5.2 Adaptive Routing Strategy

In this subsection, we introduce the adaptive-routing strategy to enhance the reliability of the 3D NoC. First, we briefly describe the Adaptive Layered Shortest Path Routing (ALASH) algorithm [119], and then, provide a detailed discussion of how adaptive routing can be implemented on top of the ALASH algorithm to improve the reliability.

6.5.2.1 ALASH Routing Algorithm

ALASH routing algorithm is built upon the idea of adaptively switching the layered shortest paths in between the source and destination pairs for any message in the network [119]. In this algorithm, the whole network is divided into a set of layers depending on the number of virtual channels available to the input/output buffers of the router, and each router is given a distinct layer identity. To ensure freedom from deadlock during the routing, messages are not allowed to re-enter any layer once they leave that particular layer. At the intermediate routers, any message can choose its own routing path, and change the layers considering the network dynamics to ensure the best shortest path available. ALASH routing algorithm is especially suitable for irregular network topologies [103] and hence, we incorporate the ALASH routing for the 3D SWNoC.

6.5.2.2 Adaptive Routing for Improving Reliability

To improve the TSV MTTF values and overall reliability of the 3D SWNoC, we modify the baseline ALASH routing algorithm and term it as *Adaptive* routing. At any instance of time, \( t \), adaptive routing considers the TSV utilization metric, \( v_n(t) \) for every VL, and identifies the VLs with relatively higher utilization. If any VL has higher average utilization metric, \( \text{avg}(v_n(t))_{for \, t=0:T} \) (here, \( T \) is the simulation duration), then its MTTF value will be relatively low. So, reducing \( v_n(t) \) at instant of time, \( t \), and repeating this periodically helps in improving the MTTF values of highly
utilized VLs. Adaptive routing strategy tracks the highly utilized VLs, and do not utilize them for certain period, and consequently, updates the necessary routing tables accordingly. As a result, the MTTF values of these highly utilized VLs improve compared to the non-adaptive routing case.

To implement adaptive routing in the 3D SWNoC, every router possesses an extra counter that keeps track of the traffic utilization of the VLs connected to it. For any region of 3D SWNoC, the Adaptive routing algorithm aims at reducing $v_n(t)$ for highly utilized VLs. This is mainly driven by the fact that if any VL fails, then the neighboring VLs in that region needs to carry extra traffic, which reduces their MTTF values. If the utilization of any VL crosses the average utilization of that region by one standard deviation amount (say a pruning rule), then the ALASH routing updates (say an update rule) the routing path by avoiding this VL. We call this overall scheme as prune-and-update, where the pruning rule is used to prune the TSVs to be considered for routing and the update rule is used to select the alternate routing paths to minimize the performance penalty.

Additionally, the EDP per message may increase marginally due to the utilization of possible non-shortest paths in those periods. However, the improvement of MTTF values for highly utilized VLs is more significant than this small performance penalty (as we will discuss in Section 6.2) and

![Figure 6.4 Illustration of ALASH based adaptive routing strategy for 3D SWNoC, and establishment of performance-reliability trade-off. In adaptive routing, the main goal is to increase the MTTF values of the highly utilized VLs incurring small EDP penalty. The parameter, $\sigma$, denotes the standard deviation in the MTTF values at time $t$.](image)

The parameter, $\sigma$, denotes the standard deviation in the MTTF values at time $t$. The overall scheme is called prune-and-update, where the pruning rule is used to prune the TSVs to be considered for routing and the update rule is used to select the alternate routing paths to minimize the performance penalty.
hence, the lifetime of the 3D NoC increases. Figure 6.5 summarizes the key ideas, implementation strategies for Adaptive routing, and the overall reliability improvement flow for 3D NoC.

6.5.3 Spare TSV Allocation Method

As discussed earlier in Chapter 4, spare TSV (s-TSV) allocation has been advocated to address the reliability concerns in a 3D IC. The main idea of this strategy is to allocate different numbers of s-TSVs for one or a group of f-TSVs such that whenever any f-TSV fails, the neighboring s-TSV is selected from that bundle to replace it [23] [28].

Following the s-TSV allocation idea, different strategies have been proposed. A 3D hybrid memory was developed, where 2 spares were allocated for a bundle of 4 f-TSVs, and signal switching technique was exploited to select the appropriate spares [73]. Similarly, a signal shifting based technique was developed with 1:N (mainly N was selected to be 4) spare to f-TSV ratio in [23] [149]. Additionally, there are other improved methodologies for spare allocation as well [150] [22] [151]. All these works mainly explore the trade-off between additional TSV resources and the degree of fault tolerance. As the ratio of spare to the f-TSV increases, the fault tolerance capacity of 3D NoC increases.

In this work, we also consider s-TSV allocation strategy by focusing on the workload-induced failures of the TSV enabled VLs. As a case study, we consider the spare allocation methodology following the signal switching technique from [73], where 50% additional TSVs were allocated for four f-TSVs. It is to be noted that we consider this particular spare TSV allocation scheme for comparative performance and reliability trade-off analysis only and thus, similar analysis can be performed with other spare allocation algorithms as well. Figure 6.5(a) illustrates the spare TSV
allocation with 50% additional spares from [73]. For a 3D NoC, we consider 50% additional TSVs for a vertical links (consists of 8 functional TSVs). Figure 6.5(b) illustrates the spare TSV allocation for vertical links in a 3D NoC with the same approach.

Importantly, allocation of spares improves the MTTF values of f-TSVs up to a certain limit, however, this process doesn’t necessarily improve the energy efficiency or temperature profile of a 3D manycore chip. Additionally, when a TSV-enabled VL run out of spares, and some other f-TSVs fail, then the functionality of that VL will be compromised. In the worst case, that VL will be considered as a failed one and as a result, overall EDP of the system increases. Hence, the energy profile of spare allocation initially (at time $t = 0$) remains same as the original 3D NoC, and the improvement in reliability of the 3D manycore chip comes at the expense of additional resources.

### 6.6 Experimental Results and Analysis

In this section, we first describe the experimental setup adopted in this work. Subsequently, we analyze the performance of different reliability improvement methodologies, and perform a comprehensive comparative study among them.
6.6.1 Experimental Setup

Our system consists of 64 cores and 64 network routers equally partitioned in four layers. We have employed the full-system simulator, GEM5, to obtain the detailed processor and network-level information [152]. To evaluate the performance of different NoCs, we use a cycle-accurate NoC simulator that can simulate any regular or irregular 3D NoC architecture [103].

The NoC routers are synthesized from an RTL level design using the ST 28-nm SOI process in Synopsys™ Design Vision. The NoC simulator uses wormhole switching protocol. For regular 3D MESH NoC, XYZ-dimension order-based routing is used. For irregular small-world based network, the topology-agnostic Adaptive Layered Shortest Path Routing (ALASH) algorithm is adopted [119]. The energy consumption of the network routers was obtained from the synthesized netlist by running Synopsys™ Prime Power, while the energy dissipated by wireline links was obtained through HSPICE simulations.

We consider four SPLASH-2 [122] benchmarks, namely, FFT, RADIX, LU and WATER, and four PARSEC benchmarks [123], namely, DEDUP, VIPS, FLUIDANIMATE (FLUID) and CANNEAL in this performance evaluation.

We consider four VFI clusters while imposing a minimum VFI cluster size of four cores. By using the K-means clustering algorithm as mentioned above [107], we clustered cores with high levels of communication running similar workloads. For voltage-frequency scaling, we consider nominal range of operation in the 28 nm technology node avoiding the turbo boost or sub-threshold regimes. Hence, the adopted DVFI uses discrete V/F pairs that maintain a linear relationship. Hence, the following V/F pairs are considered: 1.0 V/3.0 GHz, 0.95 V/2.75 GHz, 0.9 V/2.5 GHz, 0.85 V/2.23 GHz, 0.8 V/1.94 GHz, 0.75 V/1.64 GHz, 0.7 V/1.33 GHz, and 0.65 V/1.02 GHz [Chen et al. 2015].
To calculate the overall system energy, we incorporate both the regulator switching energy and DVFI controller energy.

We have also incorporated the synchronization delay associated with the mixed-clock/mixed-voltage FIFOs at the boundaries of each VFI into the simulations [153]. For each VFI, we have incorporated the energy overhead of generating additional clock signals [154], the voltage regulator’s switching, and the mixed-clock/mixed-voltage FIFOs.

Table 6.1: VFI configurations with number of cores (and routers) and static V/F levels for VFI-cluster

<table>
<thead>
<tr>
<th>Benchmarks</th>
<th>VFI 1 (#Cores-V/F levels)</th>
<th>VFI 2 (#Cores-V/F levels)</th>
<th>VFI 3 (#Cores-V/F levels)</th>
<th>VFI 4 (#Cores-V/F levels)</th>
<th>Average V/F levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFT</td>
<td>29 – 0.9</td>
<td>23 – 1.0</td>
<td>7 – 0.9</td>
<td>5 – 0.9</td>
<td>0.8694</td>
</tr>
<tr>
<td>RADIX</td>
<td>37 – 1.0</td>
<td>19 – 0.9</td>
<td>4 – 0.9</td>
<td>4 – 0.8</td>
<td>0.8403</td>
</tr>
<tr>
<td>LU</td>
<td>32 – 0.8</td>
<td>24 – 1.0</td>
<td>4 – 0.6</td>
<td>4 – 0.9</td>
<td>0.8352</td>
</tr>
<tr>
<td>CANNEAL</td>
<td>22 – 0.6</td>
<td>22 – 1.0</td>
<td>16 – 0.6</td>
<td>4 – 0.9</td>
<td>0.9089</td>
</tr>
<tr>
<td>FLUID</td>
<td>40 - 0.9</td>
<td>16 – 1.0</td>
<td>4 – 0.7</td>
<td>4 – 0.9</td>
<td>0.9233</td>
</tr>
<tr>
<td>DEDUP</td>
<td>40 - 0.9</td>
<td>16 – 1.0</td>
<td>4 – 1.0</td>
<td>4 – 1.0</td>
<td>0.8336</td>
</tr>
<tr>
<td>VIPS</td>
<td>30 – 0.7</td>
<td>26 – 0.9</td>
<td>4 – 0.7</td>
<td>4 – 0.9</td>
<td>0.8362</td>
</tr>
<tr>
<td>WATER</td>
<td>33 – 0.8</td>
<td>23 – 1.0</td>
<td>4 – 0.9</td>
<td>4 – 0.7</td>
<td>0.8815</td>
</tr>
</tbody>
</table>

In order to allow the DVFI controller to save significant energy without large performance penalties, we allow a 5% execution time penalty. In this work, we ensure that the DVFI controller delay is less than 1% of the V/F switching period, T, so that the controller doesn’t impose significant delay. From VHDL synthesis, the delay of controller is found to be 8.8 ns, and subsequently, we set the lower bound T ≥1 μs. Then, we sweep the switching window period T throughout the range 1ms ≥ T ≥ 1 μs and choose the value of T that minimizes the EDP. The optimal values of T were found to be between 9 μs and 227 μs depending on the benchmark.
Table 6.1 shows the VFI configuration for the 64-core system considered in this work. The number of elements in each cluster vary from four to forty-eight depending on the benchmark. Additionally, the static V/F levels for all the clusters are also tabulated for each benchmark. The final column shows the normalized values of the V/F levels averaged over the duration of the execution of the benchmarks.

Finally, after profiling the core and network power, to perform the thermal characterization, we employed the Hotspot simulator [155]. The average system temperature was considered for the reliability analysis.

6.6.2 Performance of 3D SWNoC with DVFI and Adaptive Routing

In this section, we evaluate the performance of 3D SWNoC with VFI-based power management and adaptive routing methods.

To evaluate the performance of 3D SWNoC, we consider the energy delay product per message (EDP) as the relevant metric. Here, we consider the network latency per message as the delay while the energy consumption per message is regarded as the energy metric. All the EDP values are normalized with respect to a 3D MESH NoC (without incorporating any VFI-based power management or adaptive routing). Figure 6.6 shows the normalized EDP profiles of 3D SWNoC with adaptive routing (Adaptive) and VFI based power management (DVFI).
with DVFI- and Adaptive routing. For ease of understanding, these configurations are referred as: 

- **3D MESH** (3D MESH without DVFI control or adaptive routing), 
- **3D SWNoC** (3D SWNoC without DVFI control or adaptive routing), 
- **Adaptive** (3D SWNoC with adaptive routing), 
- **DVFI** (3D SWNoC with VFI based power management) for all the discussions in the following sections.

From Figure 6.6, we can see that 3D SWNoC always shows significantly lower EDP values when compared to 3D MESH. In 3D SWNoC, the small-world based connectivity ensures lower average hop count. Thus, it reduces both the network latency and energy consumption. On average, 3D SWNoC shows 26% lower EDP when compared to 3D MESH. Next, we evaluate the performance of the 3D SWNoC with VFI-based power management and adaptive routing. From Figure 6.6, we can see that 3D SWNoC with adaptive routing (**Adaptive**) shows on average ~5% higher EDP values than the baseline SWNoC. This increase in EDP happens for all the benchmarks considered in this work. For adaptive routing strategy, the idea is to redistribute the total traffic of the TSV-enabled VLs in such a way that any particular VL is not disproportionately utilized when compared to others. Redistribution of traffic may create longer paths when compared to the shortest possible ones, potentially increases the average path length, and hence, the EDP per message increases for the **Adaptive** routing. Interestingly, the worst-case performance penalty for **Adaptive** remains within ~7% when compared to the baseline SWNoC. This happens due to the inherent robustness of the small-world architecture against link failures. Prior work has shown that the 3D SWNoC can tolerate as high as 15% vertical link failure by increasing the EDP by only 9% in the worst case [20]. Hence, during adaptive routing, even if the top 15% highly utilized VLs are kept unutilized, the performance penalty remains within a tolerable margin. As a result, the EDP with **Adaptive** increases only marginally.
From Figure 6.6, it is also seen that $DVFI$ shows on an average 13% lower EDP profile when compared to the baseline $3D\ SWNoC$. This reduction in EDP is expected from the VFI-based power management strategy: the VFI-controller lowers the V/F levels depending on time-varying computation and communication characteristics of each VFI cluster. As a result, the V/F levels are reduced for each cluster compared to the $SWNoC$. Hence, $DVFI$ always show lower EDP profile when compared to $SWNoC$ or the $Adaptive$ routing counterpart.

6.6.3 Thermal Profile of 3D SWNoC with DVFI and Adaptive Routing

In this section, we evaluate and analyze the thermal profiles of 3D SWNoC with adaptive routing and VFI-based power management. For evaluation purpose, we consider two metrics, viz. the average and the maximum temperatures of the system.

Figure 6.7(a) and (b) show the average and maximum temperature profiles of the 3D SWNoC ($SWNoC$), and with adaptive routing ($Adaptive$) and VFI-based power management ($DVFI$) methodologies. From these figures, we can see that both the maximum and average temperatures are reduced with the $DVFI$. The highest and the lowest temperature profiles of 3D $SWNoC$ are

![Figure 6.7(a)](image-a.png)  
![Figure 6.7(b)](image-b.png)
seen for RADIX and VIPS benchmarks respectively. The reason is that these two benchmarks are
characterized by very high (RADIX) and very low (VIPS) traffic injection rates respectively.
Compared to the baseline 3D SWNoC, the average temperature reduction with VFI based power
management is $\sim 11^\circ$C considering all the benchmarks. The best improvement of $\sim 16^\circ$C is
observed for WATER benchmark while the lowest improvement is found to be $\sim 4.5^\circ$C for DEDUP
benchmark. Similarly, the highest improvement in the maximum temperature profile is observed
to be $22^\circ$C for WATER benchmark while the average improvement for all the benchmark is $\sim 15^\circ$C.
For the $DVFI$, both the core- and network power are reduced significantly as both average voltage
and frequency levels are lowered compared to the baseline 3D $SWNoC$.

For $Adaptive$, neither the voltage nor frequency levels are reduced compared to the baseline
$SWNoC$. However, during adaptive routing implementation, the network power increases
marginally (as seen from Figure 6.6) while the core power is almost unchanged. As a result, the
temperature profile of the $Adaptive$ increases by a small amount (remains almost same) when
compared to $SWNoC$. The overall effect is that the temperature profiles of the $SWNoC$ and $Adaptive$
show similar values while the $DVFI$ is able to significantly reduce both the average and the
maximum operating temperature of 3D SWNoC.

In subsequent sections, we discuss how this reduction in temperature is translated to the
improvement in reliability of the TSV based VLs and that of the whole 3D SWNoC.

6.6.4 Improvement of MTTF with DVFI and Adaptive Routing Methodology

The MTTF values of the TSVs (and TSV enabled VLs) depend on three main factors namely- (i)
the utilization metric ($v_n(t)$), (ii) the average voltage and current levels for VLs, and the resultant
energy consumption profile, and (iii) finally, the temperature profile of the system. In this section,
we will discuss how the VFI based power management and the adaptive routing methodologies help in improving the MTTF value of the TSVs and that of the TSV-based VLs.

To evaluate the MTTF profiles of TSVs, we consider in total four representative benchmarks from SPLASH-2 and PAESEC suites namely CANNEAL, FFT, DEDUP, and VIPS. These benchmarks are characterized with wide variation of computation and communication characteristics, and their individual message injection rates are combination of high (CANNEAL), medium (FFT and DEDUP) and low (VIPS). Hence, they can be considered as the representatives for these benchmark suites.

Figure 6.8 (a)-(d) show the MTTF distribution profiles of 3D SWNoC (SWNoC) with VFI based power management (DVFI) and adaptive routing strategy (Adaptive). The horizontal axis indicates the normalized MTTF values while the vertical axis represents the percentage of occurrences for a particular MTTF. The MTTF values are normalized with respect to the lowest value for each respective benchmark. Note that the most important part of the MTTF distribution is the lower end part of the curve (critical MTTF distribution), where the VLs have MTTF values less than the median of the distribution. The VLs corresponding to the critical MTTF distribution fail earlier when compared to the other VLs and negatively affects the neighboring VLs also. Hence, this part of the MTTF distribution mainly affects the lifetime of the system and can be considered as the main concern. Moreover, the MTTF distribution having higher value compared to the median MTTF doesn’t necessarily affect the final lifetime determination, and hence, in some cases, some upper end tail of the MTTF distribution is discarded for better visualization in Figure 6.8(a)-(d). Additionally, the envelope of the MTTF distributions are plotted in the inset for each case to show
how the distributions with different reliability improvement methodologies are shifting on the MTTF axis.

From these figures, we can see that the MTTF distributions of the *DVFI* and *Adaptive* have lower percentage of occurrences (most of the cases are zero occurrences) for the lowest MTTF value bar (and for other low MTTF bars as well). The envelope of the MTTF distributions for *DVFI* and *Adaptive* also shift towards the right-hand side on the horizontal axis. From the reliability perspective, it means that both the *DVFI* and *Adaptive* reduce the number of most critical VLs.

![MTTF distribution profiles](image_url)

**Figure 6.8** The MTTF distribution profile of TSV enabled vertical links (VLs) for 3D SWNoC with VFI based power management and adaptive routing strategy for four benchmarks viz. (a) CANNEAL, (b) FFT, (c) DEDUP, and (d) VIPS. The horizontal axis indicates the normalized MTTF values while vertical axis refers to percentage of occurrence for a particular MTTF value. The MTTFs are normalized with respect to the lowest MTTF value of the system for each respective case.
which have the lowest MTTF values in the system. Importantly, when the most critical VLs fail, then they affect the neighboring VLs negatively by increasing their traffic load. Hence, the MTTF values of these critical VLs affect the NoC lifetime significantly and any increase in their MTTF values can improve the reliability of the overall system. Additionally, we can see that while the whole MTTF distribution of the DVFI shifts towards far right, but the MTTF distribution curve for the Adaptive is shifted towards the median from both the lower and upper ends and becomes skewed.

For the Adaptive, the main idea is to improve the MTTF values of the highly utilized TSV-based VLs by reducing their utilization. However, the total amount of traffic in the system remains almost same with- and without the adaptive routing strategy. Hence, to reduce the utilization of highly utilized VLs, other neighboring VLs carry more traffic. As a result, the MTTF values of some VLs (mainly those had very large MTTF values) are reduced. So, the MTTF distribution with the Adaptive tends to be skewed around the median. Importantly, neither the voltage nor the frequency is reduced for the VLs during adaptive routing. In addition, the temperature profile of the Adaptive is almost same as the baseline SWNoC. Hence, the only improvement in the MTTF distribution is achieved by the redistribution of traffic and homogenization of the utilization metric.

For DVFI, the MTTF distribution shifts towards far right on the MTTF scale when compared to both SWNoC and the Adaptive. To understand this drastic improvement in MTTF values for the DVFI, we again consider the basic factors that affect the MTTF value of the TSVs. For VFI-based power management strategy, the average traffic utilization metric, $v_n(.\text{avg})$, for each VL remains almost the same. However, the V/F levels and hence, the net amount of current is reduced for all the VLs across all the benchmarks. The MTTF value depends inversely on the amount of current
passing through the VLs ($MTTF \propto i^n$). For example, only 5% reduction in current can result in as high as 10% improvement in MTTF value. Moreover, with the $DVFI$, temperature of the system and that of the VLs is reduced on an average from $8^\circ C$ to $14^\circ C$. MTTF values of the TSVs depend on the temperature exponentially. Any reduction in temperature will significantly increase the MTTF value. As a result, with $DVFI$, the MTTF distribution improves from both the reduction in workload-induced stress resulting from lower average V/F levels, and the lowering of overall system temperature. Hence, the whole curve shifts towards far right on the MTTF scale. For all the benchmarks, the lowest MTTF value for $DVFI$, increases by 2.5 to 4 times compared to that of the baseline SWNoC.

In what follows, we discuss how this improvement in MTTF values with the VFI based power management and the $Adaptive$ routing methodologies is translated to the overall lifetime improvement of the 3D SWNoC.

### 6.6.5 Lifetime Improvement of 3D SWNoC

The lifetime definition of 3D SWNoC adopted in this work is the instant of time when the EDP of 3D NoC equals to that of 3D MESH. The parametric value of lifetime indicates how long the 3D NoC performs better than 3D MESH.

In previous sections, we have mainly analyzed the performance and the MTTF distribution profile of 3D SWNoC with VFI-based power management and $Adapting$ routing strategy. However, in addition to these two reliability improvement techniques, another popular method is the allocation of spare TSVs as discussed in Section 2 and Section 5. In addition to the $DVFI$ and $Adaptive$ routing-based reliability improvement techniques, we also consider the spare TSV allocation for 3D SWNoC. As described in Section 5.3, we follow the signal switching technique based spare
allocation policy from [73], where for a bundle of four functional TSVs, two additional TSVs are provided as spares. This allocation scheme can tolerate failure of any of the two TSVs within the bundle. We consider this spare TSV allocation policy and the resultant performance of 3D SWNoC with spare allocation is marked as $sTSV_{4:2}$ for the rest of this work.

Figure 6.9 shows the normalized lifetime of the $SWNoC$, $DVFI$, $Adaptive$, and $sTSV_{4:2}$. For comparative analysis, all the lifetime values are normalized with respect to the lifetime of the $SWNoC$. From the figure, we can see that with the $DVFI$, $Adaptive$, and $sTSV_{4:2}$, the lifetime of 3D SWNoC increases as expected. However, the amount of improvement is different for each case. Among all the reliability improvement methodologies, the $Adaptive$ and $sTSV_{4:2}$ show similar amount of lifetime improvements over the baseline $SWNoC$, while drastic increase in lifetime is observed for the $DVFI$. For example, with $WATER$ benchmark, the lifetime is improved by more than 5 times with the $DVFI$.

To explain the improvement in lifetimes of different reliability enhancement methodologies, we need to consider two performance metrics for 3D NoCs, viz., the EDP profile and the MTTF distribution of the VLs.

First, we analyze the performance of the $sTSV_{4:2}$ allocation methodology from EDP and MTTF distribution profile perspectives. For the $sTSV_{4:2}$ allocation technique, the sTSVs are not used initially. Hence, the EDP profile of $sTSV_{4:2}$ doesn’t change compared to the $SWNoC$ at time $t=0$ (at the starting point), as in both the cases, the initial NoC configurations are the same. However, whenever any f-TSV fails, the available spares replace the failed f-TSVs, and consequently, the NoC can maintain the same EDP profile with $Spares_{4:2}$ allocation methodology for longer period of time. For $SWNoC$, the EDP starts to increase from the instant of time any f-TSV fails. In a 3D
NoC, a bunch of f-TSVs form a VL. For the \textit{Spares\_4:2} allocation methodology, the ratio of spares to the functional TSVs is 50\% (considered in this work). So, if more than 50\% of f-TSVs fail in any VL, then that particular VL runs out of spares, and the VL can be considered as a failed one.

The net result of the \textit{Spares\_4:2} allocation strategy for 3D SWNoC is that the overall MTTF for any particular VL increases due to allocation of spares, and the EDP profile remains unchanged for a longer period of time compared to the \textit{SWNoC}. Hence, the lifetime of 3D SWNoC with the \textit{sTSV\_4:2} methodology increases. However, due to the limited number of spares, the overall lifetime improvement is not significant. For all the benchmarks considered in this work, the average lifetime improvement achieved with the \textit{Spares\_4:2} is 19\% while the maximum and minimum values are 22\% and 8\% for FLUID and CANNEAL benchmarks respectively.

The \textit{Adaptive} routing methodology for 3D SWNoC also shows same order of magnitude improvement in lifetime as the \textit{Spares\_4:2}. The average lifetime improvement for all the benchmarks is 27\% for \textit{Adaptive}, while the \textit{Spares\_4:2} achieves 19\% improvement. The maximum and minimum improvements are achieved with \textit{Adaptive} routing for LU and RADIX benchmarks respectively. For the \textit{Adaptive}, EDP increases marginally compared to the baseline \textit{SWNoC} due to re-routing of messages in an attempt to homogenize the utilization metrics, $v_n(t)$.

![Figure 6.9 Normalized lifetime of 3D SWNoC with VFI based power management, adaptive routing strategy, and with 4:2 spare TSV allocation. All the lifetime values are normalized with respect to the lifetime of the baseline 3D SWNoC.](image-url)
However, the MTTF values are improved for all the benchmarks for the *Adaptive* as seen from Figure 6.8(a)-(d). Hence, the lifetime of 3D SWNoC also increases but remains comparable with the *Spares_4:2* allocation methodology.

We now consider the lifetime profile of DVFI-based power management methodology for 3D SWNoC, which shows drastic lifetime improvement over the other methodologies. For the *DVFI*, we again consider the EDP profile and MTTF distribution of the VLs to analyze the lifetime improvement. With VFI-based power management, due to the reduction of the V/F levels, on an average the EDP reduces by 13% compared to the baseline *SWNoC*. Additionally, due to the reduction of the operating temperature, MTTF values of the VLs improve significantly. From Figure 6.8(a)-(d), we can see that the MTTF values are improved by at least a factor of 2.5 (for FFT) to 3 (for CANNEAL) for the most critical VLs. These improvements, both in the EDP profile and the MTTF values, result in significant lifetime improvement for the *DVFI*. The average improvement achieved for all the benchmark is 2.35 times, while the maximum and minimum lifetime improvements are achieved for DEDUP and WATER benchmarks respectively. This improvement in lifetime is significantly more than that achieved through adaptive routing and spare allocation. These significant improvements are mainly achieved due to the reduction of operating temperature enabled by power management.

The effects of VL utilization metric, V/F levels, workload-induced stress, system temperature, and the overall EDP profile on the lifetime improvement for 3D SWNoC are summarized in Table 6.2. The comparative changes (compared to the *3D SWNoC*) in respective reliability improvement factors for the *Spares_4:2, Adaptive, and DVFI* are marked with up or down arrow. From this table, we can see the cumulative benefits of *DVFI* to increase the lifetime of the 3D SWNoC.
Table 6.2: Summary of reliability improvement factors for a 3D SWNoC

<table>
<thead>
<tr>
<th>Lifetime Improvement Criteria</th>
<th>Spares_4:2</th>
<th>Adaptive</th>
<th>DVFI</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDP Value</td>
<td>–</td>
<td>↑</td>
<td>↓↓</td>
</tr>
<tr>
<td>Avg. utilization metric, $v_{n_{avg}}$</td>
<td>–</td>
<td>for critical VLs, ↓</td>
<td>–</td>
</tr>
<tr>
<td>Through current, $i$</td>
<td>–</td>
<td>–</td>
<td>↓</td>
</tr>
<tr>
<td>System temperature</td>
<td>–</td>
<td>–</td>
<td>↓</td>
</tr>
<tr>
<td>TSV MTTF values</td>
<td>–</td>
<td>for critical VLs, ↑</td>
<td>↑↑</td>
</tr>
<tr>
<td>Overall lifetime of system</td>
<td>↑</td>
<td>↑</td>
<td>↑↑</td>
</tr>
</tbody>
</table>

* Here, – indicates no or small change when compared to 3D SWNoC.
Similarly, ↑ and ↓ denote increase and decrease in respective value while ↑↑ and ↓↓ indicate large increase and decrease in value.

6.6.6 Implementation Overhead for Reliability Improvement Methodologies

In this section, we evaluate the implementation overheads of all the reliability improvement methodologies considered in this work and perform a comparative analysis. The additional logic circuits required for the spare allocation and adaptive routing methodologies, and the VFI controller are synthesized using Synopsys design compiler in ST 28nm libraries. Importantly, the energy overhead for all the reliability enhancing methodologies considered in this work are already included for the performance analysis presented in Sections 6.1 to 6.5. Here, we put the details of implementation overhead for each methodology.

Table 6.3: Summary of implementation overheads for reliability improvement methodologies

<table>
<thead>
<tr>
<th></th>
<th>Spares_4:2</th>
<th>Adaptive</th>
<th>DVFI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional total area (mm²)</td>
<td>0.16</td>
<td>0.158</td>
<td>0.093</td>
</tr>
<tr>
<td>Additional energy per message</td>
<td>6.32 pJ</td>
<td>32 pJ</td>
<td>0.948 pJ</td>
</tr>
</tbody>
</table>

To implement the Spares_4:2 allocation scheme for 3D SWNoC, we need 50% spare TSVs for each VL, and the corresponding selection switches [73]. The pitch for the TSVs considered in this
work is 20 μm [Plas et al. 2011]. The selection switches are implemented with multiplexer and de-
multiplexers. The total area overhead for the spare TSVs is 0.16 mm$^2$. The selection logic circuitries
on both ends of the TSV bundle cost 6.32 pJ extra energy for each communicated message.

For Adaptive routing methodology, every router possesses an extra counter to keep track of the
number of flits that passed through the associated VLs. Next, this information is utilized to update
the routing tables. The flit numbers are used to find out the highly utilized VLs, and subsequently,
for determining which VLs should not be utilized for the next period of time. Note that the routing
tables already exist in each of the routers, and hence, the overhead of these tables are not included
for the Adaptive. The total area overhead for Adaptive is found to be 0.158 mm$^2$ and the extra
energy consumed for each message is 32 pJ.

For the DVFI implementation, we implement the feedback based VFI controller. The VFI
clustering methodology is an offline and one-time process. To represent the worst case, the DVFI
controller is implemented considering all the 64 cores and routers to be clustered in a single group.
The total area overhead for the worst-case scenario is 0.093 mm$^2$ and the energy consumption
overhead per message is 0.948 pJ.

Table 6.3 presents the implementation overhead associated with the Spares_4:2, Adaptive, and
DVFI methodologies. If we consider a planar die area of 10x10 mm$^2$, then the overall area overhead
for all the methodologies remain within 1%. The highest area overhead is found to be for the
Spares_4:2 allocation strategy due to the large area of a single TSV. While the area overhead of
the DVFI and Adaptive are associated with the logic circuits, however, for the spare allocation
methodology, this mainly comes from the additional TSVs. These additional TSVs also impose
fabrication and packaging challenges when compared to the \textit{DVFI} and \textit{Adaptive} reliability improvement methodologies.

6.7 Summary of TSV-enabled 3D NoC Reliability Analysis

Workload-induced stress is one of the main factors behind failure of TSV-based vertical links in a 3D NoC. In this work, we analyze how the workload-induced stress affects the MTTF value of different vertical links, and the overall performance and lifetime of a 3D small-world network-based NoC (SWNoC). To improve the reliability of such a 3D NoC, we incorporated VFI-based power management technique for manycore chips in an attempt to simultaneously improve both the energy efficiency and reliability. The DVFI-enabled 3D SWNoC on an average achieves 39\% better EDP compared to conventional 3D MESH NoC, and significantly improves the MTTF values of the highly utilized vertical links. Furthermore, we benchmark the performance of VFI-based power management with respect to two existing reliability improvement methodologies, viz. adaptive routing and spare TSV allocation. The DVFI-enabled 3D NoC enhances the lifetime of the 3D SWNoC significantly more than that achieved via either adaptive routing or spare TSV allocation. All the reliability improvement strategies considered in this work do not introduce any noticeable area and energy overheads. Considering the performance and reliability trade-offs, and implementation overheads, we found VFI based power management as the most promising strategy to enhance the reliability of a 3D NoC.
7 MONOLITHIC 3D INTEGRATION FOR 3D NOC DESIGN

7.1 Motivation for Monolithic 3D Integration for NoC Design

Monolithic 3D (M3D) integration, a breakthrough technology to achieve “More Moore and More Than Moore,” opens up the possibility of designing cores and associated network routers using multiple tiers by utilizing monolithic inter-tier vias (MIVs) and hence, reducing the effective wire length [156] [33]. Compared to TSV-based 3D ICs, M3D offers the “true” benefits of vertical dimension for system integration: the size of an MIV used in M3D is over 100x smaller than a TSV [35] [34] [36]. This dramatic reduction in via size and the resulting increase in density opens up numerous opportunities for design optimizations in 3D NoCs. Figure 7.1(a) shows an illustrative example of monolithic 3D integration and its cross-sectional view. Figure 7.1(b) compares the TSV-based integration with M3D.

Figure 7.1 (a) Illustration of Monolithic 3D (M3D) and TSV-based 3D structure. While two thick layers i.e. bonding layer and die substrate are present between the tiers for TSV-based integration, however, for M3D-enabled architecture, only thin inter-layer dielectric (ILD) layer is present. (b) Comparison of physical dimensions of TSV and MIV vias. The figure is redrawn from [84].
Motivated by the promise of M3D and the potential of 3D NoCs that use MIVs, we propose a design methodology for M3D-enabled high performance and energy-efficient NoC architectures. The proposed approach explores the possibility of designing an efficient NoC enabled by true multi-tier routers. We also undertake a detailed performance evaluation between M3D- and TSV-enabled 3D NoCs.

7.2 Design Space of Monolithic 3D Integration-enabled NoC

In this section, we formally introduce the M3D-enabled NoC design space. Since a TSV-based 3D SWNoC outperforms other existing counterparts [16], we consider it as a baseline in this paper to demonstrate the benefits of M3D-integration.

7.2.1 NoC Design with M3D Integration

High-density integration in M3D is enabled by the alignment precision of MIVs, which has been reported to be 10 nm for the 22-nm technology node [157] [82] [83]. Hence, the typical number of MIVs, in general, is significantly higher compared to TSV-based designs [86].

In an NoC, the network routers form the communication backbone. The total network energy consumption consists of the link and router energies. In order to reduce the energy consumption and network latency of the 3D NoC, the benefits of M3D integration can be exploited. More specifically, the routers can be designed to extend over multiple tiers. In this context, the benefits of transistor- and gate-level partitioning offered by the M3D integration can be utilized for multi-tier logic and NoC router design [30] [83] [85] [158].

The benefits of extending the routers over multiple tiers are three-fold. First, the intra-router wirelength and the energy consumption are reduced. Second, the inter-router hop count of the
network reduces as the routers extended over multiple tiers can communicate with other routers without traversing planar wires. By designing the routers over two tiers in M3D-integration for a 3D MESH NoC, we can save 30% in footprint, and 16% in energy per message compared to their TSV counterpart [7]. Third, the TSVs that consume high energy are replaced with energy efficient MIVs. While the energy consumption of MIVs are comparable to that of planar metal wires, the associated energy per bit for the TSVs are higher. Depending on the implementation methodology and technology parameters, the values of the typical TSV energy per bit are 4~5 times more than that of metal wires [84] [83].

To illustrate the benefits of an M3D SWNoC over the TSV-enabled counterpart, Figure 7.2 shows a conceptual view of TSV- and M3D-based 3D SWNoCs. As seen from this figure, in the M3D SWNoC, in addition to the single-layer routers (blue), some routers are extended over multiple layers (orange). In contrast, for a TSV-based design, all the routers are extended only over a single-layer (blue). To explain how this helps in improving the NoC performance, we consider the inter-node communications for the source-destination pair (0, 59) as an example. The associated SW-graph connectivity is shown in the middle part of Figure 7.2. The path highlighted with green
indicates the shortest available path for this particular pair. The shortest-path (SP) lengths for M3D- and TSV-based SWNoC are calculated to be 3 and 6 hops respectively, which shows significant reduction in hop count for the M3D configuration. Additionally, we get improvements in energy as MIVs are more energy-efficient compared to TSVs. In the next section, we explore how to exploit these benefits to design an energy-efficient 3D NoC architecture.

7.2.2 Small-world network-enabled M3D NoC

A small-world (SW) network lies in between a regular, locally interconnected mesh network and a completely random Erdős-Rényi topology. SW graphs are characterized by very short average path length, defined as the number of hops between any pair of nodes. The average shortest path length of SW graphs is bounded by a polynomial in $\log(N)$, where $N$ refers to the number of nodes \[104\] \[15\]. To design a SW-enabled NoC, we follow the power-law based connectivity \[103\]. The probability of establishing a connection between two nodes varies exponentially with the link-length e.g. $p(l) \propto l^{-\alpha}$, where $p(l)$ is the probability of connecting two nodes with link length, $l$, and the parameter $\alpha$ is defined as the connectivity parameter of the network. Essentially, $\alpha$ determines the amount of irregularity introduced in the network and the overall link distribution.

In a SWNoC architecture, the achievable performance largely depends on how efficiently the network is optimized for a given set of performance metrics \[17\]. In this context, both the placement of the routers and links (horizontal and vertical) are crucial for achieving high performance and energy efficiency. For the router placement, the important factors are the multi-tier placement and their respective locations in the M3D NoC design space. On top of this, we need to place the horizontal and vertical links optimally such that the average utilization of the network resources (routers and links) is minimum for any kind of data exchange. Minimum
resource utilization ensures considerable reduction in average wirelength and less energy consumption.

7.2.3 Problem Setup

In this section, we describe how we exploit the benefits of M3D- integration to design a small-world network enabled architecture.

7.2.3.1 Cost Function

To optimize any NoC architecture, we define a unified cost function, $O$, that combines all the relevant performance metrics. Our target is to achieve the lowest possible value for this cost function. The cost function is termed as the communication cost and defined as –

$$O = \sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} (r \cdot h_{ij} + l_{ij}) \cdot f_{ij}$$

(7.1)

where $N$, $h_{ij}$, $l_{ij}$ and $f_{ij}$ denote the total number of nodes, the hop count, the minimum link length, and the frequency of interaction between the $i^{th}$ and $j^{th}$ nodes respectively. The parameter $r$ refers to the routing stages including the virtual channel allocation, input- and output-port arbitration, path allocation inside a router. Optimizing the cost function, $O$, ensures lower average hop count, (near-) optimal link and router placements to ensure reduced resource utilization. Subsequently, both the router and link energy consumptions reduce, and improves the overall NoC performance.

7.2.3.2 Problem Statement

The NoC design space consists of a set of cores, $C = \{C_1, C_2, \ldots, C_N\}$, where $N$ refers to the total number of cores and the associated routers. Each core is connected to a nearby router and the routers are connected using the power law model. In addition, the whole design space consists of $P$ tiers, and the vertical communications are enabled via MIVs. The link distribution of the network
is denoted by $L$, where $L = \{l_1, l_2, \ldots, l_t\}$ and $l_i$ denotes the number of links having a particular link length $i$. Here, the parameter $t$ indicates the maximum link-length present in the network.

The performance of the M3D SWNoC depends on the effectiveness of the joint placement of the routers and links in the NoC design space. To optimize the link and router placements, we aim to achieve the lowest value of the communication cost of the network, $O$ as defined in (1). The physically feasible solution space of the M3D SWNoC is combinatorial in nature. Hence, any kind of optimization algorithm viz. simulated annealing (SA) or genetic algorithm (GA), can be applied to find the optimized configuration. However, in our approach, we employ a machine-learning (ML)-inspired methodology as it is shown to outperform SA- and GA-based solutions [17]. Optimizing the cost function minimizes the average wire-length traversed by any message. Overall, any message spends less time in the network and, thereby, reduces both router and link energy consumptions.

For a fair comparison of the optimized M3D SWNoC with other existing NoC architectures, we constrain the SWNoC to have the same number of links as a 3D MESH. We also undertake a comparative performance evaluation of various 3D NoC architectures designed using M3D. In addition, we establish the design trade-offs considering the TSV- and M3D-based integrations.

7.2.3.3 Network Constraints

To design the M3D SWNoC architecture, we need to incorporate the physical constraints imposed by the M3D technology. For the traditional TSV-based designs, we could only allow the placement of vertical links point-to-point (regularly) between routers placed at different planar layers. In addition, the logic blocks needed to design the routers are constrained to be placed in a single planar die in the 3D space. On the other hand, for M3D-based integration, routers are extended
over multiple tiers. However, even with the current state-of-the-art fabrication technology, if any logic block is extended over more than two-tiers, the performance varies among different tiers due to mismatch in transistor speed and interconnect-material (e.g., Copper (Cu) and Tungsten (W) are used in two different tiers) [159] [160]. Hence, we limit the M3D-routers to be extended only over two consecutive tiers.

Figure 7.3 Illustration of the design space and optimization complexity comparison between TSV- and M3D-integration-based small-world (SW) network-enabled 64-node 3D NoC architectures.

7.2.4 Optimization Complexity and Challenges

It is important to note that with multitier cores and routers in the M3D SWNoC, the set of candidate link locations is much larger than the corresponding TSV-based solutions where cores and routers are only placed in specific planar layers. Therefore, the M3D SWNoC optimization problem is a significant generalization of the TSV-based optimization: we need to optimize over the joint space of link locations and the corresponding M3D SWNoC designs. For example, the size of the candidate set for the location of links is \((N/P)!\) and \((N)!\) in TSV and M3D based designs respectively, where \(N\) is the number of cores and \(P\ (>1)\) is the number of planar layers in TSV-based design. For a high number of cores (~1000 or more) and a finite number of planar layers in TSV-based 3D IC, the difference in the sizes of candidate link location sets is enormous. It can be easily shown that the search space is at least \((P\cdot N)\) times larger for M3D, e.g., an intimidating
31000 times for $N = 1000$ and $P = 3$. Figure 7.3 shows an illustrative example for the design and optimization space of the TSV- and M3D-enabled 3D SWNoC architectures.

### 7.2.5 Other 3D NoC Architectures for Performance Evaluation

For the comparative performance evaluation among different 3D NoCs other than 3D MESH, we consider two other recently proposed irregular NoC architectures, namely mrrm and rrrr [57]. For mrrm, the links and routers are placed completely randomly in the middle two tiers while the other two tiers are designed with the regular mesh-based interconnection. In the rrrr NoC, links and routers are placed completely randomly in the four tiers. We also employ the M3D-integration based designs for these NoCs.

Figure 7.4 Instantiation of STAGE-based 3D SWNoC architecture optimization approach enabled with monolithic integration. The base search, $A$, optimizes the objective function, $O$, while meta search, $M$, guides the base search to initial solution space that has promise to lead better quality solution. The evaluation function, $E$, is a regression learner that learns the design space of M3D-enabled NoC.

### 7.3 Optimization of M3D SWNoC Design

In this section, we explain the NoC design optimization methodology that employs machine learning to handle the challenge of searching the extremely large M3D design space, which is
combinatorial in nature with all possible candidate link and router placements. The key idea is to intelligently explore the design space to improve the computational-efficiency for finding near-optimal designs. This approach was recently shown to be very effective for TSV-enabled 3D SWNoC optimization [161]. In this work, we employ this algorithm for M3D-enabled SWNoC optimization. Additionally, we qualitatively discuss the effect of different design parameters.

7.3.1 ML-based NoC Design Optimization Approach

To find the optimized NoC architecture, we employ an online learning algorithm called STAGE, which was originally developed to improve the performance of local search algorithms (e.g., hill climbing) with random-restarts for combinatorial optimization problems [105]. The accuracy of local search algorithms critically depends on the quality of the starting states. The key insight of STAGE is to use the past problem-solving experience (local search runs) to learn an evaluation function $E$ that can predict the quality of a starting state and use it to select good starting states. The accuracy of $E$ improves as the algorithm explores more designs via local search runs.

STAGE is an iterative algorithm. Figure 7.4 shows a high-level illustration of the key computational steps performed in each iteration for M3D SWNoC optimization. We initialize $E$ (evaluation function) with a random function. In each iteration, we first run a local search procedure $A$ guided by $E$ (meta search) to select a good starting state. Subsequently, we run local search $A$ from the selected starting state guided by the original objective function $O$ (base search) until reaching a local optimum. If the prediction of evaluation function $E$ is not correct, we generate new training data to improve $E$. At the end of each iteration, the aggregate training data is given to a regression learner to update $E$. As the iterations progress, the accuracy of $E$ will improve and it can be leveraged to explore more promising parts of the design space. We employed regression
tree learning algorithm from the WEKA toolkit [118] to learn $E$ for NoC design optimization. The algorithm returns the best M3D SWNoC design that is uncovered during search for a given time bound or maximum number of iterations.

7.3.2 Effects of Design Parameter Variation on the 3D NoC

In this section, we discuss the effects of different NoC design parameters on the optimization methodology and overall NoC performance.

7.3.2.1 Effects of Connectivity Parameter, $\alpha$, on M3D SWNoC

The connectivity parameter, $\alpha$, determines the link distribution, $L$, for a SWNoC. For a fixed total number of links $M$, a smaller value of $\alpha$ introduces comparatively more long-range links than short-range links. Consequently, a smaller value of $\alpha$ helps in improving the performance of long-distance communication. Following similar arguments, a larger value of $\alpha$ improves the performance of local communication. Hence, an optimum value (or a range of values) of $\alpha$ exists for which the SWNoC achieves the best performance. It has already been shown that for TSV-based 3D SWNoC, the optimum value of $\alpha$ is 2.4 [16]. However, for the M3D SWNoC, the multi-tier router placement adds a new dimension to the network optimization. Hence, the effects of the parameter $\alpha$ on an M3D SWNoC need to be evaluated.

7.3.2.2 Effects of Link Reduction

Reducing the total number of links in the network increases the average hop count and communication cost. Hence, the network latency and energy consumption of the NoC increase. However, at the same time, the average connectivity per router also decreases, which helps in minimizing the energy consumption per router. For the SW-network enabled NoC, the average hop count increases only marginally when any link is removed [104] [15]. With the possibility of
extending the router to multiple tiers in M3D NoC, the increase in average hop count is even lower with any kind of link removal. Hence, if the M3D SWNoC is designed and optimized efficiently, it can still potentially perform better than the MESH-based counterpart with less resources. This hypothesis needs to be evaluated experimentally.

### 7.3.2.3 Effects of Introducing Additional Routers in a 3D NoC

The energy consumption of a router depends on the routing mechanism, and the size of the cross-bar connecting the input and output ports. For a fixed number of links in the NoC, if the number of the routers is increased, then the average port count per router is reduced. Consequently, the cross-bar size inside the router is reduced as well. This should result in lowered energy consumption per router. However, introduction of additional routers also increases the network diameter which can potentially degrade the NoC performance. Hence, a trade-off point exists for the number of additional routers and the overall NoC performance. The key challenge of introducing more routers is to determine the number of additional routers that achieves optimum NoC performance, and to place the routers optimally (along with all the horizontal and vertical links) in the NoC design space (due to more number of routers compared to the number of cores).

### 7.4 Results and Analysis

In this section, we present the performance evaluation of M3D-enabled NoC architectures considered in this work. For this performance evaluation, we consider three metrics: latency, energy dissipation, and energy-delay-product (EDP). We define the EDP as the product of network latency and energy dissipation to unify the effects of both these two metrics into a single parameter.
7.4.1 Experimental Setup

To evaluate the performance of different NoCs, we employ a cycle accurate NoC simulator that can simulate any regular or irregular 3D architecture. Our system consists of 64 cores and 64 network routers equally partitioned in four tiers with each tier having an area of 10x10 mm². For the TSV-based 3D NoCs, we consider a 4:1 serialization ratio for enabling the most energy efficient vertical communication [162]. The physical dimensions of the TSVs are 2.3 μm x 15 μm (diameter x length) [26]. Similarly, the respective dimensions for the MIVs are 50nm and 100 nm [21]. The length of each packet is 64 flits and each flit consist of 32 bits. All router ports have a buffer depth of two flits and each port has four virtual channels. We use wormhole routing for data exchange. For regular 3D mesh-based NoC, XYZ-dimension order routing is used. For irregular architectures, such as the SW network, the topology agnostic Adaptive Layered Shortest Path Routing (ALASH) algorithm is shown to be very effective [119]. The NoC routers were synthesized using Synopsys. The energy dissipation of the wireline links was obtained through HSPICE simulations, taking into consideration the length of the wireline links. It should be noted that as the MIVs are as small as local vias, using standard 2D cells in Synopsys to synthesize the NoC routers does not add any additional overhead [30].

For performance evaluation of different 3D NoCs, we consider four SPLASH-2 benchmarks [122], namely, FFT, RADIX, LU, and WATER, and four PARSEC benchmarks, namely, DEDUP, VIPS, FLUIDANIMATE (FLUID), and CANNEAL [123], in this performance evaluation. These benchmarks are chosen as they widely vary in communication and computation characteristics. In addition, the individual traffic injection rates also vary from low (VIPS) to high (CANNEAL, RADIX).
7.4.2 Evaluation of M3D SWNoC Optimization

In this section, we compare the performance of the STAGE algorithm with respect to simulated annealing (SA) when applied to the M3D SWNoC design space. It is to be noted that although we have considered the M3D SWNoC as the testbed for characterizing the performance of the optimization algorithms, other NoC architectures can be used as well.

7.4.2.1 Best Cost NoC Design: STAGE vs SA

To evaluate the performance of the optimization algorithms, we consider two metrics, viz. the quality of the solution and the convergence time. The value of the objective function (communication cost in (7.1)) at any instant of time indicates the quality of the optimized NoC. A lower value of the cost at any instant of time indicates that a better solution has been explored by the optimization algorithms. Figure 7.5(a) shows the best achieved cost for STAGE- and SA-based optimization algorithms as a function of time. As a case study, we have considered the average traffic injection rates of all the benchmarks considered (SPLASH-2 and PARSEC) in this work.

Figure 7.5 (a) Best-achieved communication cost for the M3D SWNoC during the link and router placement optimization with STAGE- and simulated annealing (SA)- algorithms. (b) Convergence time improvement for STAGE-based link- and router-placement optimization over SA-based approach for M3D- and TSV-enabled SWNoC architecture.
As can be seen from the figure, STAGE reaches the (near-) optimal value much quicker than the SA-based approach. This happens due to two main reasons. For the STAGE-based approach, we employ a guided search, which exploits all the knowledge acquired from past exploration of the design space to guide the base search in subsequent iterations. It also employs the features of the candidate NoC design and the learned evaluation function to select highly promising candidate solutions and, thereby, discards a large portion of the candidate space that does not necessarily improve the performance of the optimization algorithm. On the other hand, the SA approach probabilistically improves the quality of the solution in each iteration, where each iteration itself also takes a long time to finish due to the large number of candidate solutions in the search space.

Finally, the convergence time for each algorithm is also benchmark-dependent. Hence to complete the analysis, Figure 7.5 (b) shows the normalized convergence time for STAGE and SA. As STAGE takes less time in each case, the runtimes are normalized with respect to STAGE for each benchmark. It is seen that STAGE-based optimization performs significantly better than the SA-approach. The improvement factor for STAGE ranges from 8 to 15 depending on the benchmark.

![Figure 7.6](image_url) (a) Normalized network latency, (b) EDP per message (w.r.t. M3D MESH) of random and optimized M3D SWNoC.
7.4.2.2 Performance of M3D-enabled NoC: Random vs. Optimized

In this section, we compare the performance of the optimized M3D SWNoC with a randomly generated SW network. Figure 7.6(a) and Figure 7.6(b) show the energy consumption and EDP for random and optimized M3D SWNoC, marked as \texttt{SW\_random} and \texttt{SW\_optimized} respectively. For ease of comparative study, all the metrics are normalized with respect to the M3D MESH NoC.

From the figure, it is seen that for all the metrics, the performance of the M3D SWNoC improves after optimization. The maximum improvement is achieved for the CANNEAL benchmark (14% reduction in energy consumption and 17% reduction in EDP), while the lowest reduction in energy and EDP is seen for FLUID (3% and 4% respectively). The \texttt{SW\_optimized}, on an average (averaged over all the benchmark considered in this work), shows 14% and 15% lower values for energy consumption and EDP respectively compared to \texttt{SW\_random}. To explain this, Table I shows the communication cost for the \texttt{SW\_random} and \texttt{SW\_optimized} NoCs for different benchmarks under consideration. As seen from the table, for all the benchmarks, this metric is

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Communication cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>\texttt{SW_random}</td>
</tr>
<tr>
<td>FFT</td>
<td>5.62</td>
</tr>
<tr>
<td>RADIX</td>
<td>14.52</td>
</tr>
<tr>
<td>CANNEAL</td>
<td>6.77</td>
</tr>
<tr>
<td>LU</td>
<td>1.57</td>
</tr>
<tr>
<td>DEDUP</td>
<td>3.11</td>
</tr>
<tr>
<td>FLUID</td>
<td>1.53</td>
</tr>
<tr>
<td>WATER</td>
<td>1.25</td>
</tr>
<tr>
<td>VIPS</td>
<td>1.22</td>
</tr>
<tr>
<td>Average</td>
<td>5.07</td>
</tr>
</tbody>
</table>

Table 7.1: Average hop count and best achieved communication cost for the optimized M3D SWNoC
reduced significantly for the optimized M3D NoC compared to a randomly generated topology. In addition, the average hop count also decreases after link and router placement optimization. As a result, the network latency, energy, and EDP for the SW_optimized improve.

7.4.3 Scalability Analysis for NoC Optimization

As discussed in Section 3.3.4, the number of candidates in the design space for the SWNoC increases exponentially with an increase in system size. Consequently, the design space exploration becomes more challenging and the convergence time of the adopted optimization algorithm increases as well. To explain this more elaborately, we perform the convergence time analysis for the adopted STAGE-based optimization technique for different system sizes viz. 64, 128, and 256. For other larger system sizes beyond 256, the convergence time is extrapolated by using the best fitted polynomial curve from the available experimental data. In this analysis, we are more interested in comparative performance improvement of STAGE-based NoC optimization compared to the SA algorithm. Hence, the convergence times shown in Figure 7.7(a) and Figure 7.7(b) are normalized with respect to the respective values of STAGE for different benchmarks.

It is to be noted that only for this comparative performance analysis, we have considered a different set of benchmarks from the MCSL suite [163] and undertook network-level analysis instead of GEM-5 based full-system performance evaluation. The GEM-5 based full system performance evaluation using SPLASH-2 and PARSEC benchmark can only be undertaken with a system size up to 64 cores. For larger system sizes, e.g. 128, 256, 512, we need to depend on network level simulations. Moreover, the MCSL benchmark suite extracts realistic traffic patterns of some real applications by executing a cycle-accurate simulator, and also the on-chip traffic patterns can be obtained for any system size. As a result, we consider the MCSL-benchmarks suites for
experimenting with larger system sizes. In total, we consider, seven applications from this suite namely- FFT, FPP, H264, ROBOT, RS_dec, RS_enc, and SPARSE [163]. Finally, we consider the average value of the convergence time for all these applications. Figure 7.7(a) shows the average normalized convergence time for STAGE- and SA-based link and router placement optimization algorithms. We have experimentally evaluated the convergence times of STAGE- and SA- algorithms for three different system sizes viz. 64, 128, and 256. From these data, the improvement factor in convergence time has been extrapolated for bigger system sizes and shown in Figure 7.7(b).

From Figure 7.7(a), it is seen that the STAGE-based NoC optimization technique has significantly lower runtime compared to the SA counterpart. For three different NoC system size of 64, 128, and 256, the SA-optimization takes 12, 23, and 95 times more time to converge compared to the STAGE-based approach. As an example, for the 256-core system, while the STAGE takes 4~5 hours, the SA-based optimization takes more than two-weeks to converge to the final solution. It is seen from Figure 7.7(b), that SA takes ~350 times more runtime compared to the STAGE-based approach for the 512-core system. It can be shown that even with a conservative estimation, the

![Figure 7.7](image_url)

(a) Normalized runtime of STAGE and SA algorithm for NoC link and router placement optimization, (b) Normalized runtime of SA for different NoC size with extrapolation for larger system.
runtime with SA can extend over months. This is expected to increase further as the system size grows beyond 512.

7.4.4 Comparative Performance Evaluation of M3D SWNoC

In this section, we evaluate the performance of the M3D SWNoC architecture with respect to other 3D NoCs viz. 3D MESH, mrrm, rrr as described in Section 3.3.5. For fair comparison among the 3D NoC architectures, all the NoCs are designed with the same number of links. For comparative performance evaluations, we normalize the performance metrics of each NoC with respect to the M3D MESH. Figure 7.8(a) and Figure 7.8(b) show the normalized energy consumption and EDP respectively.

As seen from the figures, among all the M3D NoC architectures considered in this work, the M3D SWNoC performs better than other NoCs while the MESH NoC performs the worst. The mesh-based NoCs suffer from multi-hop communications and hence, shows the highest energy and EDP values. The other two architectures namely mrrm and rrrr perform in between the MESH and SWNoC. On average, the SWNoC shows 29% and 32% lower energy and EDP respectively compared to 3D MESH counterpart. The maximum improvement is achieved for CANNEAL
(35% improvement in energy and 39% improvement in EDP), while the lowest improvement (18% in energy and 20% in EDP) is obtained for the FLUID benchmark. For the M3D SWNoC, the link distribution follows the power law and the interconnection architecture is optimized to facilitate both the local and the long-range communications. In addition, the routers are extended over multiple tiers and their respective placements are also optimized by finding the best cost M3D SWNoC design.

On the other hand, both mrrm and rrrr have some kind of small-world network effects in the horizontal planes with random link placements. However, their routers are placed and extended over multiple tiers randomly and hence, they are not optimized suitably. As a result, they do not fully exploit the benefits of either the SW network or M3D-integration and, hence, perform in between the M3D MESH and optimized SWNoC architecture.

![Figure 7.9](image)

Figure 7.9 Normalized (a) energy per message, (b) EDP per message of an optimized 3D SWNoC with TSV- and M3D-enabled designs (normalized w.r.t. TSV-based design).

### 7.4.5 Evaluation of TSV- and M3D-based NoCs

In this section, we evaluate the performance of M3D SWNoC with its TSV-based counterpart. Figure 7.9(a) and Figure 7.9(b) show the normalized energy and EDP for both NoCs. We normalized all the values with respect to that of the TSV-based 3D NoC.
As seen from the figures, the M3D SWNoC achieves 28% and 30% lower energy and EDP respectively compared to the TSV-based design. To explain this, Table 7.2 shows the average hop count and communication cost for these two architectures for the CANNEAL benchmark as an example. As seen from the Table, the values for both these parameters are reduced for M3D SWNoC compared to the TSV-based design. In addition, the wirelength in M3D-based designs are reduced significantly, and the MIVs also dissipates lower energy compared to the TSVs. As a result, both the energy and EDP profiles improve for M3D-enabled NoC.

### 7.4.6 Effects of Different SWNoC Design Parameters

In this section, we analyze the effects of different SWNoC design parameters on the overall performance. As EDP combines both the latency and energy consumption profile of the NoC, we consider this unified metric for the performance analysis in this section. We consider mainly three design parameters, viz., the SW connectivity parameter ($\alpha$), expanding NoC router logics to multiple tiers exploiting the benefits of M3D integration, and the total number of horizontal and vertical links.

#### 7.4.6.1 Performance Variation with Connectivity Parameter, $\alpha$

The connectivity parameter, $\alpha$, determines the degree of small-worldness of the M3D SWNoC. It has been shown that for a network with the dimension, $T$, the parametric value of $\alpha$ being less than

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Optimized 3D SWNoC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TSV-enabled</td>
</tr>
<tr>
<td>Average hop count</td>
<td>2.97</td>
</tr>
<tr>
<td>Communication cost</td>
<td>6.10</td>
</tr>
</tbody>
</table>

Table 7.2: The best achieved average hop count and communication path length for the TSV- and M3D-enabled optimized SWNoC.
|T+1|, ensures the small-world characteristic \([104] \ [101]\). Hence, to analyze the effects of \(\alpha\) on the 3D NoC performance, we consider the range of \(\alpha\) from 1.5 to 3.6. Here, the lower bound of \(\alpha\) is not further reduced from the value of 1.5 as it was found that the performance of SWNoC does not necessarily improve for such reductions. Figure 7.10(a) shows the EDP profile of the optimized M3D SWNoC for different SPLASH-2 and PARSEC benchmarks. To show the relative performance difference for variation in connectivity parameter value, \(\alpha\), the EDP is normalized with respect to the maximum EDP among all the benchmarks.

It is seen from the figure that for different benchmarks, the M3D SWNoC achieves the minimum EDP for different values of \(\alpha\). However, for any particular benchmark, the performance of the optimized SWNoC does not vary widely with \(\alpha\). In fact, the difference between the highest and the lowest EDP values remains within \(\sim 6\%\). This happens mainly for two reasons. Firstly, during the optimization, for any given value of \(\alpha\), the placements of links and routers are optimized to achieve the best performance. For the M3D-enabled design space, even for a given link distribution (determined by the \(\alpha\) value), the router placements (to be extended over multi-tier or not) can be varied to induce some amount of small-world effect into the network and thereby, find low-cost configurations.
NoC designs. Secondly, as described in the Section 8.4.1, we adopt the ALASH-based routing algorithm for the irregular NoCs. During the routing, it ensures the usage of shortest path from all the available options. As a result, it always tries to achieve the best performance for any NoC configuration. Hence, the variation of performance remains within a small percentage for the M3D SWNoC architecture even for a wide range of $\alpha$ values.

To illustrate the effects of different applications and variation in connectivity parameter value on the SWNoC performance more elaborately, Figure 7.10(b) shows the average EDP values (averaged over all the benchmarks from Figure 7.10(a)). It is seen that the M3D-enabled SWNoC shows almost similar EDP value for the connectivity parameter values ranging from 2.1 to 2.7. Hence, we can choose any value from this region to design the optimized M3D SWNoC for ensuring the best achievable performance. In this work, we have chosen the optimum value of the connectivity parameter to be 2.1 for further performance evaluation of M3D SWNoC.

![Figure 7.11 Normalized (a) energy, (b) EDP per message of M3D- SWNoC with single and multi-tier routers.](image)

### 7.4.6.2 Effects of Multi-Tier Expanded Logic-block Placement

M3D-integration enables extension of router logic blocks over multiple tiers. This ultimately improves the performance of the 3D NoC. Best performance can be achieved if we extend the routers across all the four-tiers of the system. However, using the state-of-the-art fabrication
technology, the logic blocks suffer from performance variation when fabricated using multiple tiers and the variation increases with the increase in the number of tiers [159]. This will require additional synchronization schemes both for the logic blocks as well as the interconnects. Consequently, considering this technology limitation, we restrict the routers to be extended over two tiers only.

To explain the performance benefits of extending routers over two-tiers, we consider two different M3D SWNoC architectures: i) routers are extended over two tiers, and ii) routers are designed using one tier. Here, we only consider the routers to be extended over two consecutive tiers e.g. 0 and 1, 1 and 2, 2 and 3 etc. For ease of referencing, the performance of these M3D SWNoCs are marked as M3D_mulit-tier and M3D_single-tier respectively. It is to be noted that both the NoCs are optimized with the earlier mentioned optimization methodology to achieve the best performance. The energy and EDP profiles are shown in Figure 7.11(a) and Figure 7.11(b) respectively. For comparison, both these metrics are normalized with respect to the M3D MESH NoC.

From the figure, it is seen that the energy and EDP values for the M3D_mulit-tier are reduced by 4.5% and 5% respectively compared to the M3D_single-tier architecture. To explain this, we again consider the average hop count and communication cost. It is seen that for the M3D_mulit-tier architecture, both the average hop count and communication cost are reduced compared to M3D_single-tier NoC. As a result, both the energy and EDP profile improve for the M3D_mulit-tier M3D SWNoC.
7.4.6.3 Performance of SWNoC with NoC Resource Reduction

In this section, we analyze the performance of M3D SWNoC by reducing the available resources. More specifically, a certain percentage of horizontal and vertical links are removed from the M3D SWNoC architecture and then, the NoC is optimized to achieve the best performance. In total, we consider a reduction of up to 25% in the number of links. Following the previous design steps, we optimize the link and router placements for each NoC configuration with reduced resources. The performance of these M3D SWNoC architectures with different percentage of links removed are denoted as xx%, where xx is the amount of link reduction from the baseline M3D SWNoC with the same number of links as a 3D MESH NoC. Figure 7.12(a) shows the EDP profile of these different M3D SWNoC architectures with different percentages of links removed for various benchmarks where all the values are normalized with respect to the M3D MESH (without any link reduction). Figure 7.12(b) shows the average EDP considering all the benchmarks with various amount of link reduction.

As seen from Figure 7.12(a), when we remove more links, the EDP of the M3D SWNoC increases. However, even with link reduction, M3D SWNoC achieves lower EDP than the 3D MESH with
no links removed. More specifically, even if we remove 25% of the total links from the M3D SWNoC, it still performs better than the 3D MESH with 100% links. This happens mainly for two reasons. First, the SWNoC is inherently robust against any kind of link failure and hence the performance penalty arising due to reduction in number of links is very small. Second, the link and router placement optimization are performed for the M3D SWNoC with the reduced resources. The optimization approach adopted in this work, always aims to explore the best available NoC configuration from the design space. As a result, the performance degradation is minimized, and the M3D SWNoC shows better performance with significantly fewer number of links compared to the conventional M3D MESH architecture.

7.4.6.4 Performance of 3D NoC with Additional Routers

In all the above performance evaluations, we have considered a 3D NoC design space that consists of equal number of cores and routers. However, M3D-integration offers the possibility of introducing additional logic resources compared to the TSV-based integration. Hence, in this section, we explore the possible NoC performance improvement analysis with M3D-integration by adding more number of routers than the cores. As discussed in Section 4.2.3, introduction of additional routers (consequently the total number of routers becomes more than the number of total cores present in the NoC) reduces the average router size and the average port count per router. In addition, the reduction of the router and cross-bar size inside the routers result in lower router energy consumption. However, this may not necessarily lead to overall energy reduction as the network diameter increases.

To evaluate the effects of additional routers, we consider the M3D SWNoC performance with different numbers of additional routers. In this case also, with added routers, the M3D SWNoC is
optimized with the STAGE algorithm. As a case study, we introduce a maximum of 25% additional router resources with the baseline 64-core and router-enabled SWNoC. Hence, the modified NoC contains 64 cores and different number of routers ranging from 64 to 80. The performances of these architectures are denoted by SWNoC_xx, where xx indicates the number of routers present in the 64-core NoC architecture.

![Normalized average energy and EDP per message of M3D-SWNoC with different percentage of additional routers incorporated in the NoC architecture.](image)

Figure 7.13 shows the average EDP profile of an M3D SWNoC for different configurations. The performance is normalized with respect to the baseline M3D SWNoC architecture. From the figure, it is seen that as the number of routers increases, the EDP increases. This happens due to the fact that with the introduction of additional routers, the energy consumption of the comparatively bigger routers decreases, however, the overall link and router energy consumption increase due to an increase in the network diameter. The increase in energy consumption due to higher network diameter is more than the energy savings arising due to smaller routers. As a result, the overall energy consumption and network latency increase, leading to higher EDP values. Hence, we can conclude that a larger number of routers, i.e. individual routers are smaller, does not help in improving the performance of the M3D NoC architecture.
7.5 Summary of M3D-enabled 3D NoC Design

We have explored a robust and computationally-efficient design methodology for energy-efficient and reliable manycore architectures enabled by M3D-integration. We combined the benefits of a small-world network and the M3D-integration to design a M3D-enabled small-world NoC (M3D SWNoC). By adopting a machine-learning inspired approach, the links and routers were placed optimally for the proposed M3D SWNoC. We carried out a detailed performance evaluation and established relevant design trade-offs for the proposed M3D SWNoC architecture by using the SPLASH-2 and PARSEC benchmarks. We demonstrated that the optimized M3D SWNoC shows 32% lower EDP than the M3D MESH. In addition, the M3D-enabled NoC performs 28% better in terms of EDP compared to the TSV-based counterpart. Finally, the optimized M3D SWNoC outperforms a mesh-based architecture even with 25% fewer links.
8 SUMMARY OF CONTRIBUTIONS AND FUTURE RESEARCH DIRECTIONS

8.1 Summary of Contributions

Manycore processing platforms are revolutionizing the technology by increasing the computation capability of a single chip while ensuring high performance and energy efficiency. The interconnection fabrics, located at the heart of the manycore chip, dictates the overall achievable performance. In this context, 3D NoC can ensure to achieve the “More Moore and More Than Moore,” by exploiting the benefits offered by the vertical interconnects and 3D stacking e.g. improved form factor, reduced wirelength, lower capacitance value, improved energy efficiency, and higher bandwidth for internode communication.

In this work, we have analyzed and presented design methodologies for developing energy efficient and high-performance 3D NoC architectures. We proposed the design of 3D small-world NoC (3D SWNoC) architecture. By exploiting the vertical dimension in a 3D IC, we mapped the tasks among the cores in such a way that physically long distant and highly communicating cores are placed along the vertical dimension, and hence, small-worldness is ensured. To explore optimized NoC architecture, we developed machine learning based optimization algorithms for NoC link placement. We have demonstrated that the adopted STAGE-based optimization can outperform tradition simulated annealing and other algorithms in terms of quality assurance and convergence time. The proposed 3D SWNoC architecture when optimized for energy efficiency
and high performance, outperforms other regular and irregular 3D NoC architectures in terms of performance and robustness against link failures.

Following the performance analysis, we explored the reliability analysis of the chip. The vertical links of 3D ICs are predominantly designed with TSVs, which undergo workload-induced stress and fails over time. As a result, the performance of 3D IC degrades. We proposed several reliability improvement methodologies to counteract the electromigration and crosstalk noise effect caused by the wear-out of TSVs. We have explored efficient spare TSV allocation techniques, VFI-based power management, and adaptive routing to minimize the penalty due to such failures. The VFI-based power management approach has shown to improve both the energy and thermal efficiency of the architecture.

In addition, we explored the monolithic 3D-enabled NoC design space. By exploiting the benefits of reduced capacitance and improved energy efficiency of monolithic-inter-vias (MIVs), we proposed to adopt small-world architectures for NoC design. We demonstrated that M3D-based designs can outperform traditional TSV-enabled manycore chips.

Finally, we outlined some promising directions for future work based on our previously explored results and experiences. One of the major directions will be focusing on the security analysis of the NoC and overall chip. The reliability of the M3D-enabled chip need to be studied extensively as well. Finally, our future plan is to synergistically combine machine learning techniques for design, reliability and security analysis of NoC architectures.
8.2 Future Work Directions

In this section, we briefly outline the future work directions based on our findings of this work. This includes the exploration of security analysis for NoCs and reliability analysis of M3D NoCs.

8.2.1 Hardware Security Analysis of NoCs

Many-core system-on-chip (SoC) with many cores can achieve high performance and energy efficiency for both compute- and data-intensive applications. A many-core SoC integrates different types of cores (CPU and GPUs), memory, sensors, and application specific accelerators all running and coordinated by software to achieve high performance and support diverse functionality while consuming minimum energy. These SoCs are the workhorses in all electronic platforms including desktops, laptops, and smartphones.

Vendors of these electronic platforms such as Apple and Samsung have an incentive to continually design, develop, and market new versions of their platform offerings. All vendors have an incentive to entice their customers to upgrade to the new models including by downgrading the platforms that they sold. This is called planned or programmed obsolescence. Planned obsolescence is as old as the electric light bulb [164].

Apple has been sending software updates to intentionally slow down old iPhones [165], [166], [167], [168]. The French government is investigating Apple for planned obsolescence. Apple does not dispute the claim and is responding that the slowdown is necessary to address the performance degradation of the iPhone battery over time. Similarly, the Italian antitrust agencies are investigating Semiconductor giant Samsung for planned obsolescence [169].
From the NoC perspective, a key component of a 3D NoC are the vertical links (VLs) between the different layers. Most VLs use through silicon via (TSV) bundles. However, the through silicon via (TSV)-based VLs are subject to wear out due to electromigration and crosstalk noise when heavily utilized leading to failure over time [170].

Hence, a malicious program that injects a high traffic load to specific VLs will accelerate aging and ultimately fail them. This in turn will introduce significant performance penalty and energy overhead of the 3D NoC-enabled manycore chips.

This form of attack is developed by the design houses themselves for financial benefits after the warranty period is over. In this case, the netlist is already present with the design house, and hence, reverse engineering is not necessary. Also, unlike the warranty-based threat, no malicious non-expert user is required. The design house can generate the malicious program and deliver them as update packages. Once these are installed, the design house has complete access to the IC. They can now continue applying these programs to stress the TSVs and thus, reduce the lifetime of the IC, thereby forcing the user to buy a newer model.
8.2.1.1 Possible 3D NoC Attack Methodology

The main objective of an NoC attack is to accelerate the aging of the VLs in order to ensure the failure of the manycore chip as quickly as possible. An easy way to accelerate the aging of the NoC is to target the TSV-enabled VLs and increase their individual workloads.

![Diagram of NoC attack scenarios](image)

**Figure 8.1** Explanations of three different kinds of NoC attack scenario depending on the NoC domain and architectural expertise of the attacker. For uniform random attack, the hacker only needs the access to user end and manycore chip architecture knowledge. For critical region attack, the knowledge regarding the network configuration is required, while the router architecture and TSV placement configurations are necessary for critical VL attack.

The domain knowledge required to launch an attack on the NoC are depicted in Figure 8.1. We classify the accessibility of the attacker into several levels of abstractions depending on the knowledge of the whole system architecture, NoC design, router architectures and TSV location and placement.

Depending on the domain knowledge regarding the NoC, three different types of malicious programs can be designed to accelerate aging of the 3D NoC. These are- (i) uniform random attack, (ii) region-based attack, and (iii) targeted vertical link attack. In the subsequent sections, we describe the details of the respective attack, types of domain knowledge and access required for each case, and possible effects on the user end QoS.
8.2.1.2 Hardware Security Analysis of 2D Wireless NoCs

In recent years, 2D wireless NoCs has gained significant attentions due to their potential to improve the design of energy efficient manycore chips significantly. The major contributors of the wireless NoCs are the wireless channels that enable the long distant communications. For example, in a 64-core architecture, the amount of long distant communications is more than 66% and most of which are carried out by four-wireless links carry for different SPLASH-2 and PARSEC benchmarks, and graph analytics application [171] [172]. However, the wireless channels, with sophisticated engineering approaches, can be jammed [173] [174] [175] and the internode communication can suffer significant performance degradation. In addition, if the performance of the driver circuitries degrades due to NBTI and HCI carrier injections [176] [177], the effective communication bandwidth reduces. The performance of the wireless NoC undergoes significant degradations.

Hence, an external attack that targets the wireless channels and jams the communicating frequency of the chip or introduce the NBTI or HCI effects for wireless communication driver circuitries, can accelerate the aging of the chip, which in the worst case leads to failure of wireless links. The attack or planned aging of the chip manufacturer introduces significant performance degradation. From the user perspective, the chip becomes slower and he/she is forced to upgrade on the manycore chip. This is synonymous to “planned obsolescence” attack as described above. However, the target platform and chip architecture are different.

8.2.2 Reliability Analysis of Monolithic 3D NoC Architectures

In monolithic 3D integration, the logic blocks especially the routers are designed with monolithic 3D integration and consequently, it exploits the opportunity of extending the routers over multiple tiers (whenever necessary). By extending routers over multiple tiers, the average hop count reduces,
and consequently, the network latency and router energy consumption of the NoC reduces significantly. While the M3D-enabled NoC ensures high performance, however, it also faces some design challenges. These challenges include electrostatic coupling effects [34], performance degradation of bottom tier transistors [159], increased delay for upper tier interconnects [157] etc.

8.2.2.1 Electrostatic Coupling

In a M3D enabled manycore chip, tiers are placed sequentially on top of each other through the usage of a thin inter-layer-dielectric (ILD) in between the tiers. While the presence of the thin layer of ILD ensures high thermal conductivity between tiers and thermal efficient architectures, however, at the same time, it also introduces electrostatic coupling issue between the circuits from adjacent tiers [34]. The amount of electrostatic coupling of any target path depends on three parameters, viz. (i) the nature of signal transitions from adjacent logic gates or transistors placed in adjacent layers of the target path, (ii) placement and routing configurations of the metal layers below and above the target path, and (iii) the thickness of ILD layer. If the thickness of ILD is less than 30nm, which is being targeted for the upcoming M3D-enabled ICs [178], then electrostatic coupling may increase the delay of the critical most path by more than 30% [34]. In the presence of process, supply voltage, and temperature (PVT) variation, this delay is expected to increase and exceed the allowable timing budget introducing additional performance penalty.

From the NoC perspective, the routers undergo electrostatic coupling-induced delay effects and consequently, the intra-router performances are affected. The intra-router mechanism consists of multiple stages including routing computations, input- and output arbitration, virtual channel- and switch allocation, and switch traversal. In general, each of the stage is completed in one cycle under nominal conditions. However, in the presence of worst-case delay violating the timing rules,
each stage will consume one additional cycle resulting an increase of total number of cycles equals to the number of intra-routing stages delay for each message passing through the router. Consequently, the network latency and the router energy consumption per message will increase significantly for the intra-node communications. While all routers experience the electrostatic coupling-induced delay, however, the routers those are extended for multi-tiers specifically undergo the worst-case effects of signal transitions from logic gates from adjacent layers in addition to the effects of (ii) and (iii) mentioned earlier. This is because, for single-tier routers, logic blocks may not be present in the adjacent layers on top or bottom to its critical most (target) path. However, for multi-tier routers, placements of transistors or logic gates (those are undergoing different types of signal transitions during intra-routing communications) are guaranteed. As a result, the multi-tier routers experience the effects of electrostatic coupling most severely.

In addition, the switching transitions of any circuit or path depends on how frequently the logic block (router in this case) is utilized. From the NoC perspective, all routers don’t carry equal amount of traffic and hence, the switching transitions for the critical most path may vary. In general, multi-tier and large size (having more number of ports than the average) routers carry more traffics, and due to their traffic carrying high capacity, they are termed as critical routers. The critical routers are expected to have high switching transitions and more electrostatic coupling-induced effects.

### 8.2.2.2 Performance Degradation of Top-tier Transistors

As the M3D-enabled ICs are fabricated sequentially on top of each other. The operating temperature during the fabrication of top tier transistors can rise significantly high that induces damages and performance deviations for the bottom tier transistors [159]. In the worst-case, the
delay for PMOS and NMOS increases by 27.8% and 16.2% respectively [179] [159]. While the advancement of fabrication technology, may improve such performance variation, however, at the state-of-the-art designs, there remains concern over the performance of the bottom tier transistors [159]. At present, researchers are developing low-temperature process for upper tier transistors [180]. However, M3D-enabled system having more than two-tier still faces significant challenges.

From a NoC design perspective, such delay introduces additional stages for routers and/or power consumption for the intra-routing. On average, it increases the network latency. Consequently, the energy consumption and EDP per message increases resulting in significant reduction of NoC lifetime.

8.2.2.3 Interconnect Delay Variation

Similar to the performance degradation of the transistors in the bottom tier, the interconnects are also affected by the monolithic-integration fabrication process [157] [159]. Such variation in delay of the interconnects also affect the performance of the NoC between adjacent tiers and causes additional delays. In the worst-case, the communication between routers from the adjacent tiers, suffers from bandwidth differences and needs synchronizers at the boundary resulting in significant overheads.
While reducing the temperature to 625°C [180] for the upper tier fabrication in M3D, can improve the transistor, however, it is still too high for interconnects [181] [159]. One possible solution is to adopt Tungsten-based interconnect in the bottom tier, which can withstand higher temperature than the Cu interconnect [181]. However, Tungsten has higher resistivity than that of Cu (3.1 times), and hence, it can slow down the performance of the circuits. Such variation of interconnect performance between tiers can introduce significant performance penalty for the NoCs.

Figure 8.2 Different reliability concerns for Monolithic integration and reliability improvement-aware design optimization framework.

**8.2.2.4 Performance Degradation-aware NoC Design and Optimization**

The effects of electrostatic coupling, variation of transistor speed, and interconnect delay can be incorporated in the design and optimization steps of the NoC to reduce the performance degradation. The effects of the electrostatic coupling and transistor speed variation can be modeled by introducing additional routing stages. Similarly, the interconnect delay variation can be incorporated in the internode-communication during the NoC optimization.

Finally, throughout this work we have explored single objective optimization for enabling the design of high performance and energy efficient NoC. However, in the context of latency, energy
and thermal efficient NoC architectures, multi-objective optimization can be incorporated to improve the performance and reliability of the overall chip.
REFERENCES


[38] Z. Or-Bach, "A 1,000x improvement in computer systems using current fabs and process," in IEEE SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S), 2018.


