Reliable Design of Three-Dimensional Integrated Circuits

For obtaining the academic degree of

Doctor of Engineering

Department of Informatics
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany

Approved

Dissertation

by

Master of Science
Shengcheng Wang

From Tianjin, China

Date of Oral Examination: 04.05.2018
Supervisor: Prof. Dr. Mehdi Baradaran Tahoori, Karlsruhe Institute of Technology
Co-supervisor: Prof. Dr. Said Hamdioui, Delft University of Technology

Shengcheng Wang
Haid-und-Neu Str. 62
76131 Karlsruhe

Hiermit erkläre ich an Eides statt, dass ich die von mir vorgelegte Arbeit selbstständig verfasst habe, dass ich die verwendeten Quellen, Internet-Quellen und Hilfsmittel vollständig angegeben haben und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen - die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Karlsruhe, Mai 2018

Shengcheng Wang

This page would be intentionally left blank.

Abstract

Beginning with the invention of the first *Integrated Circuit* (IC) by Kilby and Noyce in 1959, performance growth in IC is realized primarily by geometrical scaling, which has resulted in a steady doubling of device density from one technology node to another. This observation was famously known as "Moore's law". However, the performance enhancement due to traditional technology scaling has begun to slow down and present diminishing returns due to a number of imminent show-stoppers, including fundamental physical limits of transistor scaling, the growing significance of quantum effects as transistors shrink, and a mismatch between transistors and interconnects regarding size, speed and power. As a result, a new paradigm shift in IC technology and architecture is required to sustain the historical IC performance growth.

Three-Dimensional Integrated Circuit (3D IC), which refers to a vertical (or horizontal) stack consisting of multiple ultra-thin IC layers, has emerged as a promising option to overcome the challenges related to conventional geometrical scaling. 3D IC can achieve device density multiplication without aggressive scaling by stacking IC layers in the third dimension. In a 3D IC, multiple IC layers can be vertically/horizontally interconnected by various interconnect techniques, in which *Through-Silicon Via* (TSV) is a critical one. More specifically, TSV is a vertical electrical connection passing completely through a silicon wafer or die, which is a key enabler for 3D integration. By using TSVs, the interconnection length between stacked IC layers can be significantly reduced compared to conventional two-dimensional counterparts. This reduction further translates into less wire delay and higher performance. However, 3D integration scheme also results in new challenges in the fields of physical design and testing mechanism. In this dissertation, a set of reliability-aware and testability-driven design and optimization techniques have been proposed in order to enable reliable designs of 3D ICs, as well as enhance physical design quality.

The first contribution of this dissertation is to develop a holistic *Computer-Aided Design* (CAD) platform for 3D *Power Delivery Network* (PDN), which can be integrated seamlessly within commercial electronic design automation development flows. The design of 3D PDN is a constrained optimization problem. An ideal PDN must: i) limit voltage drop which originates in switching circuits transients, and ii) satisfy the constraints that arise from reliability limits. In addition, these problems are further constrained by the requirement to minimize additional hardware overhead introduced by TSVs in 3D ICs. In this dissertation, a CAD platform has been developed to handle the trade-off between 3D PDN design qualities and hardware design cost under the constraints of required reliability metrics.

The second contribution of this dissertation is to develop a comprehensive framework for TSV repair in order to enhance yield and improve reliability in 3D ICs. As criti-

cal enablers for 3D ICs, TSVs can deliver signals from one layer to another. However, during fabrication and normal operations, they may suffer from various reliability issues. Due to the large number of TSVs in a chip, these issues in turn translate into low yields and reduced lifetimes. In order to fix failed TSVs, a simple but effective way is to add redundant TSVs into 3D designs to increase chip yield and overall reliability with additional hardware overheads. By considering the trade-off between chip qualities (in terms of manufacturing yield and operation lifetime), hardware overhead, and performance, a framework has been proposed to improve chip yield and extend TSV lifetime significantly with well-managed TSV redundancy.

The third contribution of this dissertation is to develop a novel test architecture for interposer-based 3D ICs (also knowns as *Two-and-a-Half-Dimensional Integrated Circuits* (2.5D ICs)), which is emerging as a precursor to TSV-based ones at the present time. In a 2.5D IC, multiple dies are placed side by side on a passive silicon interposer, and all of them must be adequately tested for product qualification. However, due to the special structure of 2.5D IC, new testing challenges have emerged such as: i) reduced number of test pins, ii) higher test power consumption, and iii) increased test-application time. In this dissertation, the research targets the above challenges, and effective solution has been developed to test dies with minimum test cost in terms of power consumption and test-application time.

As demands accelerate for increasing device density, higher bandwidths, and lower power consumption, the semiconductor community is focusing on 3D ICs, which promise "More-than-Moore" integration by packing a great deal of functionality into small form factors, while reducing cost and improving performance. However, due to several significant technical hurdles, volume production and commercial exploitation of 3D ICs are not feasible yet, and still in the early phases. For instance, the standard definitions in 3D ICs are still lacking, the supply chain ecosystem is still unstable, and design, verification and test challenges need to be resolved. In this dissertation, we focus on the reliability and testability challenges in 3D IC design. A set of reliability-aware and testability-driven techniques in the aspects of automating the physical-design process for 3D ICs have been proposed. Significant improvements for reliability and performance were demonstrated for 3D ICs. Therefore, the proposed solutions and developed frameworks in this dissertation can act as important building blocks for 3D ICs and push them toward mainstream acceptance in the near future.

Contents

List of Figures xi

List of Tables xv

List of Acronyms xvii

List of Publications xxi

1 Introduction 1
- 1.1 Benefits of 3D Integration Technologies 4
  - 1.1.1 Interconnect delay reduction 4
  - 1.1.2 Memory bandwidth improvement 6
  - 1.1.3 Heterogeneous integration 8
- 1.2 Challenges in 3D Integration Technologies 9
  - 1.2.1 Thermal management challenges 9
  - 1.2.2 3D power delivery network design challenges 10
  - 1.2.3 TSV-related reliability and yield challenges 11
  - 1.2.4 Testing challenges 11
- 1.3 Dissertation Contributions and Outline 12
  - 1.3.1 Contributions to reliable 3D power delivery network design 14
  - 1.3.2 Contributions to TSV self-repair of 3D ICs 14
  - 1.3.3 Contributions to cost-effective testing of 3D ICs 15
  - 1.3.4 Dissertation outline 16

2 Background 17
- 2.1 Interconnect Hierarchy in 3D Integration 17
- 2.2 3D Integration Technology Landscape and Applications 18
  - 2.2.1 3D system-in-package 19
  - 2.2.2 3D stacked integrated circuit 21
  - 2.2.3 3D monolithic integrated circuit 27
- 2.3 Fabrication of TSVs 27
  - 2.3.1 Via-first process 27
  - 2.3.2 Via-middle process 28
  - 2.3.3 Via-last process 29
- 2.4 Reliability Issues Related to TSVs 30
  - 2.4.1 Stresses in TSVs 30
  - 2.4.2 Electromigration in TSVs 31

2.5 Testing of 3D ICs 34

3 Thermal Stress-aware Power/Ground TSV Planning 35
3.1 Preliminaries 36
3.1.1 Power delivery network in 3D ICs 36
3.1.2 TSV-induced mobility variation 38
3.2 Motivation 39
3.3 Methodology 40
3.3.1 Problem statement 40
3.3.2 Power/ground TSV-induced delay change 42
3.3.3 Voltage constraints 43
3.3.4 Power and stress profiles redistribution 44
3.3.5 Scalability 44
3.4 Experimental Results 45
3.4.1 Experimental setup and flow 45
3.4.2 Impact of design parameters on delay variation 47
3.4.3 Comparison with previous technique 47
3.4.4 Accuracy and Runtime 48
3.5 Conclusion of Chapter 49

4 Electromigration-aware Local-via Allocation in Power/Ground TSV 51
4.1 Preliminaries and Related Work 53
4.1.1 Via-first TSV-based 3D power delivery network design challenges . 53
4.1.2 Electromigration reliability of TSVs 53
4.1.3 Electromigration-induced resistance increase of via-first TSV . . . 53
4.1.4 Related prior works 54
4.2 Proposed Electromigration-aware Local-via Allocation 56
4.2.1 Problem statement 56
4.2.2 Assumption and notation 56
4.2.3 Integer linear programming formulation 58
4.2.4 Scalability analysis 62
4.2.5 Discussion 63
4.3 Simulation Results 63
4.3.1 Simulation setup and implementation flow 63
4.3.2 Impact of temperature on local-via allocation 66
4.3.3 Impact of circuit partitioning on local-via allocation 66
4.3.4 Impact of target lifetime on local-via allocation 69
4.3.5 Comparison with prior work 71
4.3.6 Runtime analysis 73
4.4 Conclusion of Chapter 73

5 Defect Clustering-aware TSV Repair for Yield Enhancement 75
5.1 Preliminaries and Related Work 77
5.1.1 TSV fault tolerance 77

5.1.2 Spare TSV allocation in different TSV placements 77
5.1.3 TSV-induced stress and stress-induced delay overhead 79
5.1.4 TSV defect clustering 80
5.1.5 Related prior work 80
5.2 Motivational Example 81
5.3 Spare TSV Allocation Considering Defect Clustering 82
5.3.1 Problem statement 82
5.3.2 Notation 83
5.3.3 Assumptions related to metrics 83
5.3.4 Overview of the proposed methodology 85
5.3.5 Fault-map generation 86
5.3.6 Global partitioning 87
5.3.7 Detailed insertion of spare TSVs 93
5.4 Simulation Results 95
5.4.1 Simulation setup 95
5.4.2 Impact of design knobs on spare TSV allocation 97
5.4.3 Impact of process variation on spare TSV allocation 99
5.4.4 Comparison with prior work 100
5.4.5 Runtime analysis 102
5.4.6 Discussion 102
5.5 Conclusion of Chapter 103

6 Electromigration Recovery-aware TSV Repair for Reliability Improvement 105
6.1 Preliminaries and Related Work 106
6.1.1 Electromigration recovery effect 106
6.1.2 Related prior works 109
6.2 Motivation and Basic Idea 109
6.3 Design-time TSV Grouping 110
6.3.1 Electromigration-vulnerable functional TSV identification 110
6.3.2 Functional TSV partitioning 111
6.3.3 Spare TSV assignment 111
6.3.4 Non-uniform TSV grouping 113
6.3.5 Discussion 114
6.4 Run-time TSV Repair 114
6.4.1 Recovery schedule 115
6.4.2 Repair architecture 116
6.4.3 Overhead analysis 119
6.5 Numerical Results and Discussions 120
6.5.1 Experimental setup and implementation flow 120
6.5.2 Impact of design knobs on repair solution 121
6.5.3 Benefit of non-uniform TSV grouping 123
6.5.4 Comparison with conventional reactive repair approach 123
6.5.5 Comparison between static and dynamic repair approaches 125

6.5.6 Discussion . . . 126
6.6 Conclusion of Chapter . . . 127

7 Cost-efficient Multicast Testing for Interposer-based 3D ICs 129
7.1 Related Prior Work . . . 132
7.2 Motivation for Proposed Work . . . 133
7.3 Proposed Test Architecture . . . 136
7.3.1 Requirement and design . . . 136
7.3.2 Overhead analysis . . . 140
7.4 Test-scheduling Method . . . 141
7.4.1 Problem statement . . . 142
7.4.2 Notation and assumption . . . 142
7.4.3 Integer linear programming formulation . . . 144
7.5 Simulation Results . . . 146
7.5.1 Simulation setup . . . 146
7.5.2 Impact of design knobs on test schedule . . . 147
7.5.3 Comparison with prior work: BIST-based approach . . . 149
7.5.4 Comparison with prior work: scan-based approach . . . 150
7.5.5 Runtime analysis . . . 152
7.5.6 Discussion . . . 152
7.6 Conclusion of Chapter . . . 155

8 Conclusions 157
8.1 Summary of Contributions . . . 158
8.2 Future Work . . . 159
8.2.1 Leverage 3D integration for internet of things . . . 159
8.2.2 Leverage 3D integration for neuromorphic engineering . . . 160

Bibliography 161

List of Figures

1.1 Transistor count per leading edge CPU chip over last 45 years [2]. 2
1.2 (a) Gate count and wafer cost per process node leading to trend change in (b) cost per gate for different process node. Reproduced from data [8]. 3
1.3 The gap in performance measured as the difference in the time between process memory requests (for a single processor or core) and the latency of a DRAM access [9]. 4
1.4 ITRS nanoelectronics technology roadmap [10]. 5
1.5 Various 3D integration paradigms: (a) 3D stacking with wire-bonding, (b) 3D stacking with wire-bonding and flip-chip bonding, and (c) 3D stacking with TSV. Adapted from [11]. 5
1.6 The structure of 3D IC with P/G and signal TSVs [12]. 6
1.7 Relative delay for local and global interconnects and for logic gates in sub-micron technology nodes [14] 7
1.8 Comparison of clock cycles spent during computation and in memory access for three deep neural network algorithms as run on a machine learning accelerator executed in 7 nm CMOS technology. Reproduced from data by [17]. 8
1.9 Two-tier 3D IC using face-to-back TSVs 11
1.10 Proposed reliability-aware physical design and test flow for 3D ICs in this dissertation. 13

2.1 Classification of interconnects in a 3D system [36]. 18
2.2 Classification of 3D integration technologies. 19
2.3 (a) nCHIP's 3D memory stacked using gold wire-bonding [38] (b) AMKOR's 3D IC packaging stacked by copper wire-bonding [37]. 20
2.4 Top-view and cross-sectional view of the PoP for mobile DRAM and application processor [40]. 21
2.5 Cross-sectional view of the eWLB from Infineon [40]. 21
2.6 Classification of 3D integration according to different stacking approaches [47]. 22
2.7 Samsung's 64 GB DDR4 DRAM module for server farm [40]. 23
2.8 Micron's sample on HMC [53]. 24
2.9 Schematic of Hynix's HBM. 25
2.10 A passive interposer which consists of TSV and RDL supporting chips on package substrate [37]. 26
2.11 Altera/TSMC's CoWoS [58]. 26
2.12 Illustration of via-first, via-middle, and via-last TSV technology process [62]. 28

2.13 SEM images of TSVs showing centerline void after annealing at: (a) 150 °C and (b) 300 °C [81]. 31
2.14 Schematic of EM testing pattern: (a) TSV+BEOL and (b) TSV+RDL [86]. 32
2.15 TSV EM failure modes: (a) down-stream current with TSV+BEOL, (b) up-stream current with TSV+BEOL, (c) down-stream current with TSV+RDL, and (d) up-stream current with TSV+RDL [86]. 33
2.16 EM failure modes in metal line connected to TSV: (a,b) thin line with up-stream current, (c,d) thin line with down-stream current, (e,f) thick line with up-stream current, and (g,h) thick line with down-stream current [87]. 33

3.1 (a) On-chip power distribution network. (b) Three-dimensional uniform power distribution network. 37
3.2 Illustration of TSV layout parameters. 39
3.3 Carrier mobility variation surrounding TSVs. 39
3.4 Gate relocations with regular and irregular TSV topologies 41
3.5 The redistribution of gates due to the insertion of TSV. 45
3.6 The detailed implementation flow of the proposed methodology 46
3.7 Impact of KOZ and PDN pitch on induced delay variation 47

4.1 PDN of 3D ICs with via-first TSVs. 52
4.2 Illustration of a via-first P/G TSV with local vias and its equivalent resistance network. 54
4.3 Illustration of a via-first P/G TSV with local vias and its corresponding impact on signal routing. 57
4.4 Non-uniform distribution of current density within each local via in the array. Here i is the index of each via, and j is its current density. 62
4.5 Local-via distribution of P/G TSVs in *des_perf-2* and *des_perf-4* under worst case 100 ℃ and 130 ℃ temperatures. 67
4.6 Local-via distribution of the P/G TSVs in *des_perf-2* and *des_perf-4* under the same temperature. 69
4.7 The detailed local-via allocations for *des_perf-2* and *des_perf-4* under the different constraints of target lifetimes. 70

5.1 Illustration of the concept shared s-TSV technique. 78
5.2 Illustration for the motivational example. 81
5.3 Illustration of the MUX configuration for two grouping ratios. 84
5.4 Overall flow of the proposed methodology. 86
5.5 The number of unmergeable f-TSVs with varied N_{gs} and $c\%$ for benchmark b17. 98

6.1 (a) Original input driving current density. (b) Calculated EM DC equivalent current density with two different methods. 107
6.2 Comparing the nucleation time of two different methods and original stress. 108
6.3 Comparing resistance increase with AC current and average current. . . . 108

6.4 Min-cost flow problem for s-TSV assignment. 112
6.5 Illustration of the reconfigurable routing network for a (4 : 2) TSV group consisting of 4 f-TSVs and 2 s-TSVs. 117
6.6 The relationship between the achieved MTTF and $T_{\text{unit}}^{\text{stat}}$. 121

7.1 Illustration of an interposer-based 3D IC. 129
7.2 (a) Illustration for the motivational example. (b) Hierarchy in 2.5D IC testing: cores $\subset$ SoC die $\subset$ 2.5D IC. 134
7.3 Generic test architecture for multicast testing. 137
7.4 IEEE 1149.1 wrapper: (a) conventional and (b) multicast-enhanced. . . . 138
7.5 Block diagram of the proposed test architecture. 139
7.6 The relationship between fault coverage and test-application time with varying power constraints. 147
7.7 The relationship between the power budget and test-application time as F_{min} is varied. 148

This page would be intentionally left blank.

List of Tables

2.1 TSV dimensions reported in recent literature 30

3.1 Comparison with previous technique (only considering TSV-induced stress) 48
3.2 Comparison of the original delays with optimized circuit delays considering P/G TSV-induced delay changes 48

4.1 Parameter description in Equations (4.2) and (4.3). 55
4.2 Design Statistics for all benchmark. 65
4.3 Experimental setting for P/G TSV. 66
4.4 The impact of temperature on the local-via allocation in terms of optimized overflow and IR-drop ($t_{\mathrm{LT}} = 3.2$ yrs and $\Delta V^{\mathrm{avg}}_{dd,\mathrm{min}} = \Delta V^{\mathrm{worst}}_{dd,\mathrm{min}} = 0.9$). 68
4.5 The impact of circuit partition on the local-via allocation in terms of optimized overflow and IR-drop ($t_{\mathrm{LT}} = 6.4$ yrs and $\Delta V^{\mathrm{avg}}_{dd,\mathrm{min}} = \Delta V^{\mathrm{worst}}_{dd,\mathrm{min}} = 0.9$). 68
4.6 The impact of target lifetimes on the local-via allocation in terms of optimized overflow and IR-drop ($\Delta V^{\mathrm{avg}}_{dd,\mathrm{min}} = \Delta V^{\mathrm{worst}}_{dd,\mathrm{min}} = 0.9$). 69
4.7 Comparison between the proposed technique and the local vias-unaware technique [121]. 72
4.8 Comparison between the proposed technique and the routability-unaware technique [126]. 73
4.9 The runtime of the proposed methodology for all benchmarks. 74

5.1 Design Statistics for all benchmarks. 96
5.2 Yield achieved and required s-TSV count for b17 with different timing budgets. 98
5.3 Trade-off analysis between TSV yield and hardware cost for different values of N_{gs}. 99
5.4 Yield achieved and required s-TSV count for b17 with different process corners. 100
5.5 Comparison between the number of inserted s-TSVs for uniform and clustered defect distributions. 101
5.6 Comparison between the stress-unaware delay overhead and stress-aware delay overhead. 101
5.7 The runtime of the proposed methodology for all benchmarks. 102

6.1 The static recovery schedule for a $(4:2)$ TSV group. Here, A: active mode, R: recovery mode, fi: f-TSV i, sj: s-TSV j. 117

6.2 Trade-off analysis between the achieved MTTF and overhead for different grouping ratio. . . . 122
6.3 Comparison between the uniform and non-uniform approaches when targeting the same MTTF value of f-TSV network $MTTF_{target}$. . . . 124
6.4 Comparison between the proposed proactive approach and the conventional reactive approach [173]. . . . 125
6.5 Comparison between the proposed static and dynamic proactive repair approaches. . . . 126

7.1 Summary of timing and area overheads analysis 141
7.2 Design Information for three benchmarks 146
7.3 Design Data for Benchmarks Derived by Commercial Tools 147
7.4 Fault coverage achieved and test-application time for the benchmarks using [214] and the proposed technique, when the test clock frequency is assumed to be 50 MHz. . . . 150
7.5 Fault coverage achieved, required test power budget, and test-application time for the benchmarks using the conventional test approach proposed in [225] and the proposed technique, when the test clock frequency is assumed to be 50 MHz. . . . 152
7.6 Runtime of the proposed technique with varying constraints of power and fault coverage (unit: second) 152

List of Acronyms

2.5D IC Two-and-a-Half-Dimensional Integrated Circuit

2D IC Two-Dimensional Integrated Circuit

3D IC Three-Dimensional Integrated Circuit

AC Alternating Current

ATE Automatic Test Equipment

ATPG Automatic Test Pattern Generation

BEOL Back-End of Line

BIST Built-In Self-Test

C4 Controlled Collapse Chip Connection

CAD Computer-Aided Design

CMOS Complementary Metal-Oxide-Semiconductor

CMP Chip Multiprocessor

CoWoS Chip on Wafer on Substrate

CPD Critical Path Delay

CPU Central Processing Unit

CTE Coefficient of Thermal Expansion

D2D Die-to-Die

D2W Die-to-Wafer

DC Direct Current

DDR Double Data Rate

DfT Design-for-Testability

DRAM Dynamic Random-Access Memory

EDA Electronic Design Automation

EM Electromigration

eWLB Embedded Wafer Level Ball Grid Array

FEOL Front-End of Line

FF Fast-Fast

FO-WLP Fan-out Wafer-Level Packaging

FPGA Field-Programmable Gate Array

FSM Finite-State Machine

f-TSV Functional Through-Silicon Via

GDDR Graphics Double Data Rate

GPU Graphics Processing Unit

HBM High Bandwidth Memory

HMC Hybrid Memory Cube

HVM High-Volume Manufacturing

I/O Input/Output

IC Integrated Circuit

ILP Integer Linear Programming

IoT Internet of Things

IP Intellectual Property

IR Instruction Register

ITRS International Technology Roadmap for Semiconductors

KOZ Keep-out Zone

LBIST Logic Built-in Self-Test

LPD Longest Path Delay

LUT Look-up Table

MEMS Micro-Electro-Mechanical Systems

MIBLP Mixed Integer Bilinear Programming

MIC Monolithic Integrated Circuit

MIV Monolithic Inter-Tier Via

MRAM Magnetoresistive Random-Access Memory

MTTF Mean-Time-to-Failure

P/G Power/Ground

PCB Print Circuit Board

PDN Power Delivery Network

PLL Phase-Locked Loop

PoP Package-on-Package

RAM Random-Access Memory

RCP Representative Critical Path

RDL Redistribution Layer

RF Radio Frequency

SAIF Switching Activity Interchange Format

SIC Stacked Integrated Circuit

SiP System-in-Package

SoC System on Chip

SPEF Standard Parasitic Exchange Format

SS Slow-Slow

STA Static Timing Analysis

s-TSV Spare Through-Silicon Via

TAM Test Access Mechanism

TAP Test Access Port

TB Timing Budget

TSV Through-Silicon Via

TT Typical-Typical

VLSI Very-Large-Scale Integration

VP Vulnerable Path

W2W Wafer-to-Wafer

WL Wire-Length

List of Publications

This dissertation is based on the works and results presented in the following publications in print:

[1] **Shengcheng Wang**, Farshad Firouzi, Fabian Oboril, and Mehdi B Tahoori. "Stress-Aware P/G TSV Planning in 3D-ICs." In *Proceedings of Asia and South Pacific Design Automation Conference*, Jan. 2015, pp. 94-99.

[2] **Shengcheng Wang** and Mehdi B Tahoori. "Electromigration-Aware Local-Via Allocation in Power/Ground TSVs of 3D ICs." In *IEEE Transactions on Very Large Scale Integration Systems*, Jan. 2015, pp. 94-99.

[3] **Shengcheng Wang**, Mehdi B Tahoori, and Krishnendu Chakrabarty. "Defect Clustering-Aware Spare-TSV Allocation for 3D-ICs." In *Proceedings of IEEE International Conference On Computer Aided Design*, Nov. 2015, pp. 307-314 (**Best Paper Award**).

[4] **Shengcheng Wang**, Hongyang Zhao, Sheldon X.-D. Tan, and Mehdi B Tahoori. "Recovery-aware TSV Repair for Electromigration in 3D ICs." In *Proceedings of Design, Automation & Test in Europe Conference & Exhibition*, Mar. 2017, pp. 220-225.

[5] **Shengcheng Wang**, Taeyoung Kim, Zeyu Sun, Sheldon X.-D. Tan, and Mehdi B. Tahoori. "Recovery-Aware Proactive TSV Repair for Electromigration Lifetime Enhancement in 3D ICs." In *IEEE Transactions on Very Large Scale Integration Systems*, Dec. 2017, pp. 531-543.

[6] **Shengcheng Wang**, Ran Wang, Krishnendu Chakrabarty, and Mehdi B Tahoori. "Multicast Test Architecture and Test Scheduling for Interposer-Based 2.5D ICs." In *Proceedings of IEEE Asian Test Symposium*, Mar. 2017, pp. 220-225.

[7] **Shengcheng Wang**, Ran Wang, Krishnendu Chakrabarty, and Mehdi B Tahoori. "Multicast Testing of Interposer-Based 2.5D ICs: Test-Architecture Design and Test Scheduling." In *ACM Transactions on Design Automation of Electronic Systems*, April 2018, pp. 35:1-35:25.

[8] **Shengcheng Wang**, Krishnendu Chakrabarty, and Mehdi B Tahoori. "Defect Clustering-Aware Spare-TSV Allocation in 3D ICs for Yield Enhancement." In *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (In Press).

In addition, the author has completed works unrelated to this dissertation presented in the following publications in print:

[1] **Shengcheng Wang**, Farshad Firouzi, Fabian Oboril, and Mehdi B Tahoori. “P/G TSV Planning for IR-Drop Reduction in 3D-ICs.” In *Proceedings of Design, Automation & Test in Europe Conference & Exhibition*, Mar. 2014, pp. 1-6.

[2] **Shengcheng Wang**, Farshad Firouzi, Fabian Oboril, and Mehdi B Tahoori. “Deadspace-aware Power/Ground TSV Planning in 3D Floorplanning.” In *Proceedings of International Conference on IC Design & Technology*, Jun. 2015, pp. 1-4.

[3] **Shengcheng Wang**, Krishnendu Chakrabarty, and Mehdi B Tahoori. “Thermal-Aware TSV Repair for Electromigration in 3D ICs.” In *Proceedings of Design, Automation & Test in Europe Conference & Exhibition*, Mar. 2016, pp. 1291-1296.

[4] **Shengcheng Wang**, Zeyu Sun, Yuan Cheng, Sheldon X.-D. Tan, and Mehdi B Tahoori. “Leverage Recovery Effect to Reduce Electromigration Degradation in Power/Ground TSV.” In *Proceedings of IEEE International Conference On Computer Aided Design*, Nov. 2015, pp. 811-817.

1 Introduction

The development of *Integrated Circuits* (ICs) becomes indispensable in our daily life, which has allowed for the proliferation of powerful personal computing and communication solutions. All the electronic devices used constantly nowadays—from smart-phones, multimedia devices, personal computer to Internet's infrastructure, high performance computing labs, data centers—are equipped with ICs. In all of these fields, the never ending demands for higher performance, more integrated features, and increasing energy/cost efficiency drive the need for a continuous downscaling of *Very-Large-Scale Integration* (VLSI) technology. Therefore, the microelectronic industry shrinks the feature size of transistor approximately every two years, allowing the number of transistors doubles in an IC in the same time period. This prediction (or observation) became known as *Moore's law*, which was postulated by Gordon E. Moore [1] in his seminal 1965 paper [1]. It sets the electronics industry on a quest for continued scaling for more than 50 years and those who have mastered the technology have enjoyed the greatest commercial benefits. Since then, the scaling of *Complementary Metal-Oxide-Semiconductor* (CMOS) technology, following the Moore's law, allowed the semiconductor industry to successfully keep an exponential growth rate on device integration in the past decades, as illustrated in Figure 1.1.

However, evidence from recent years suggests that the trends of semiconductor scaling can no longer keep up with such a demanding scaling rate, and the current trajectory of scaling has strained the commercial sector, as much for economic as for technical reasons. While *International Technology Roadmap for Semiconductors* (ITRS) still predicts further CMOS scaling, e.g., to around 7 nm node by the year of 2020 [3], such scaling will reach fundamental physical limits. In addition, the device scaling will be slowed down even further as next generation lithography methods (e.g., extreme ultraviolet lithography and electron beam lithography) are being pushed back. On the other hand, it may not be economically beneficial to maintain the scaling even if these technological obstacles can be overcome. Figure 1.2(a) illustrates the cost per wafer for different process nodes along with the amount of usable gates per wafer due to reduced yield and effective usable area. As shown, while the fabricated gates count rises with the decreased feature size, the amount of actual usable gates during fabrication can be reduced by several issues such as defect density, leakage, doping uniformity, line edge roughness, etc. Although the cost per wafer continues to rise dramatically, the usable device count does not increase at the same rate. Consequently, the price per transistor cannot be reduced with the advanced technology node below 22 nm, and might even be reversing the trend according to some estimations (as illustrated in Figure 1.2(b)).

[1] Gordon Earle Moore (born 1929) is a co-founder of Intel, a semiconductor chip maker corporation

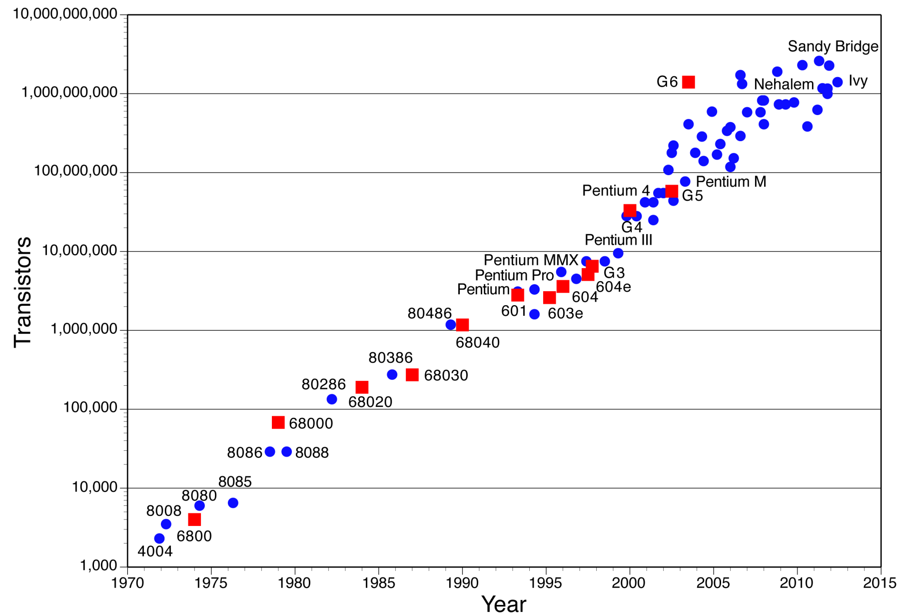
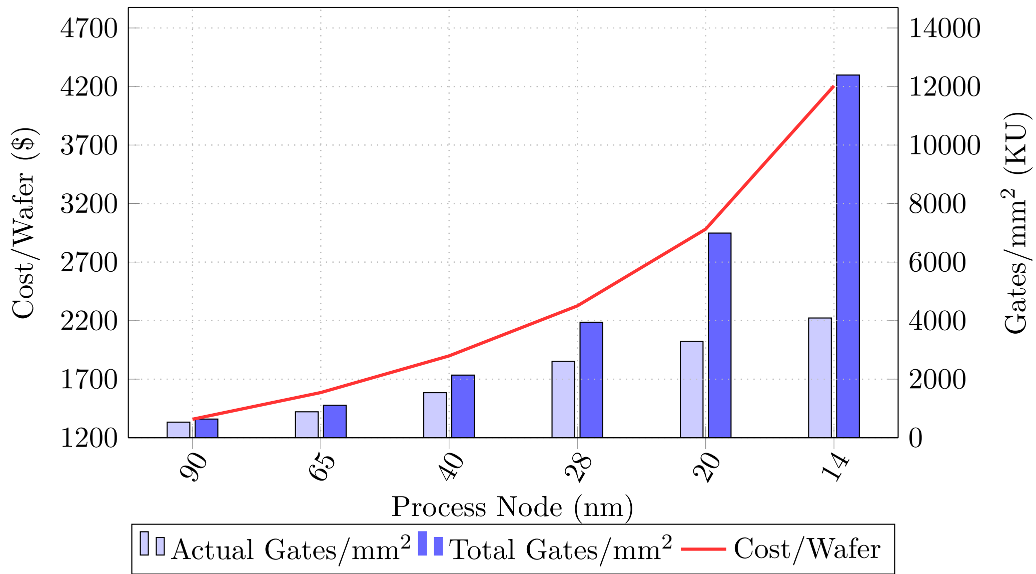


Figure 1.1: Transistor count per leading edge CPU chip over last 45 years [2].

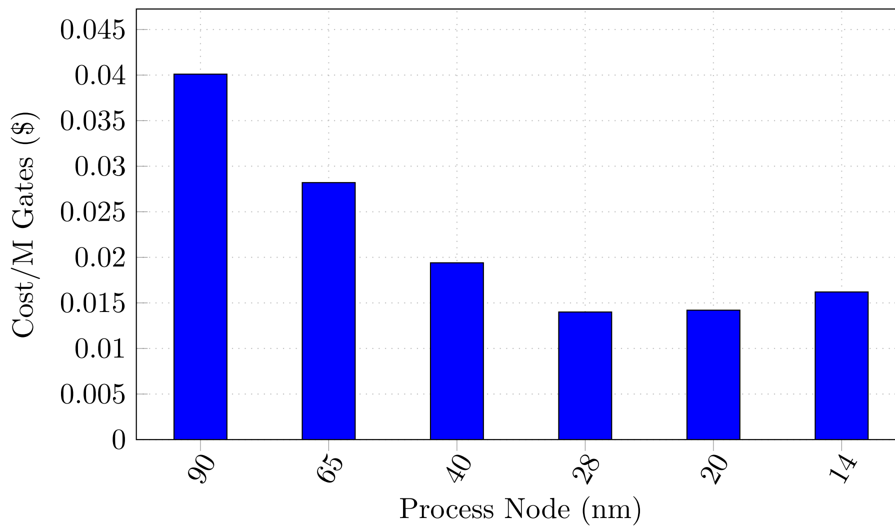
Therefore, it is impractical to continue the strive for further semiconductor scaling with the Moore's law.

As the process scaling becomes increasingly challenging, integrating more cores per chip has become the way that processors can continue to exploit Moore's law, and thus such *Chip Multiprocessor*s (CMPs) dominate today's computing landscape [4–6]. However, the increasing speed of the process is faster than that of the memory. Figure 1.3 illustrates single processor performance projections against the historical performance improvement in time to access main memory. Although single processor bandwidth has grown more slowly in recent years, the gap between *Central Processing Unit* (CPU) memory demand and *Dynamic Random-Access Memory* (DRAM) bandwidth continues to grow as the numbers of cores grow in CMPs. As a result, this imbalance between memory bandwidth and CMP data demand keeps increasing, which imposes limits on the performance achievable by even highly optimized CMPs. This growing disparity between the processor clock rate and off-chip memory access speed is popularly referred to as the "memory wall" problem [7]. In addition, this problem is further exacerbated by a lack of well-designed memory-to-logic interface. As the number of cores increases in CMPs, more and more data are needed from memory subsystem, imposing a great deal of pressure on the *Input/Output* (I/O) infrastructure. However, according to the ITRS projection, the number of pins on a package will not continue to grow rapidly enough to supply bandwidth for off-chip memory in the next decade [3], and thus new paradigms have to eventually be followed to alleviative this problem.

An alternative to further process scaling is functional diversification focusing mainly on non-digital functions via heterogeneous system integration (as illustrated in Fig-



(a)



(b)

Figure 1.2: (a) Gate count and wafer cost per process node leading to trend change in (b) cost per gate for different process node. Reproduced from data [8].

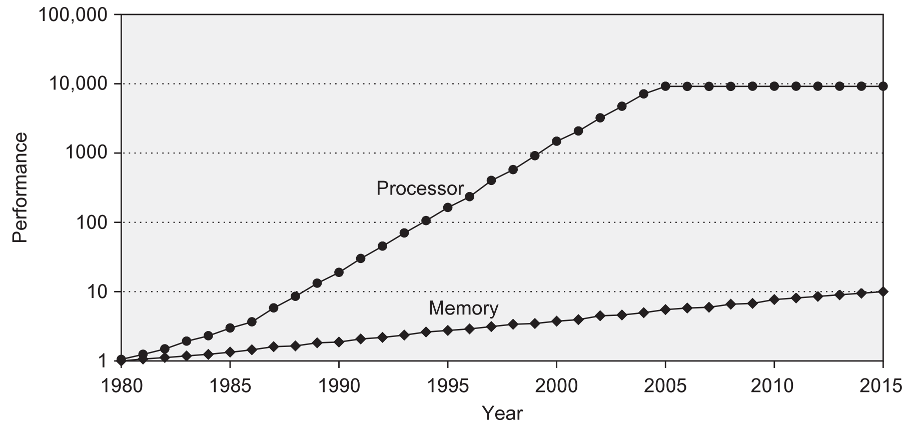


Figure 1.3: The gap in performance measured as the difference in the time between process memory requests (for a single processor or core) and the latency of a DRAM access [9].

ure 1.4), also coined as *More than Moore*. By adding higher degrees of functionality and achieving more diverse integrated system, the designers can overcome the bottleneck of the traditional down-scaling of process technology nodes and continue pushing the system performance envelope further out. In this context, and also to achieve increased packing density and shorter interconnect, the concept of *Three-Dimensional Integrated Circuits* (3D ICs) has emerged. In microelectronics, 3D ICs are integrated circuits manufactured by stacking silicon dies and/or wafers and interconnecting them together, initially by wire-bond, later by flip-chip and recently by *Through-Silicon Vias* (TSVs) or *Monolithic Inter-Tier Vias* (MIVs), as shown in Figure 2.6.

Particularly, TSV is a vertical electrical connection that passes completely through a silicon wafer or die, which is the key and most popular enabler for 3D integration. As an alternative to wire-bond and flip-chip, TSVs can achieve higher interconnection density, used to router inter-tier signals (i.e., signal TSVs) and deliver power (i.e., *Power/Ground* (P/G) TSVs), as shown in Figure 1.6. This way, the stacked dies (and/or wafers) can behave as a single device to achieve performance improvement at reduced power and smaller footprint than conventional 2D counterparts. Furthermore, 3D integration has been envisioned as a solution for future micro-architecture design to mitigated the memory wall problem by stacking memory on top of a CMP. An overview on various design approaches that leverage different benefits offered by 3D integration technologies will be presented in the next section.

1.1 Benefits of 3D Integration Technologies

1.1.1 Interconnect delay reduction

Since Jack Kilby of Texas Instruments patented the principle of integration, created the first prototype ICs and commercialized them in 1958 [13], the primary objective of IC design has been cramming more transistors onto a single chip to achieve increasingly

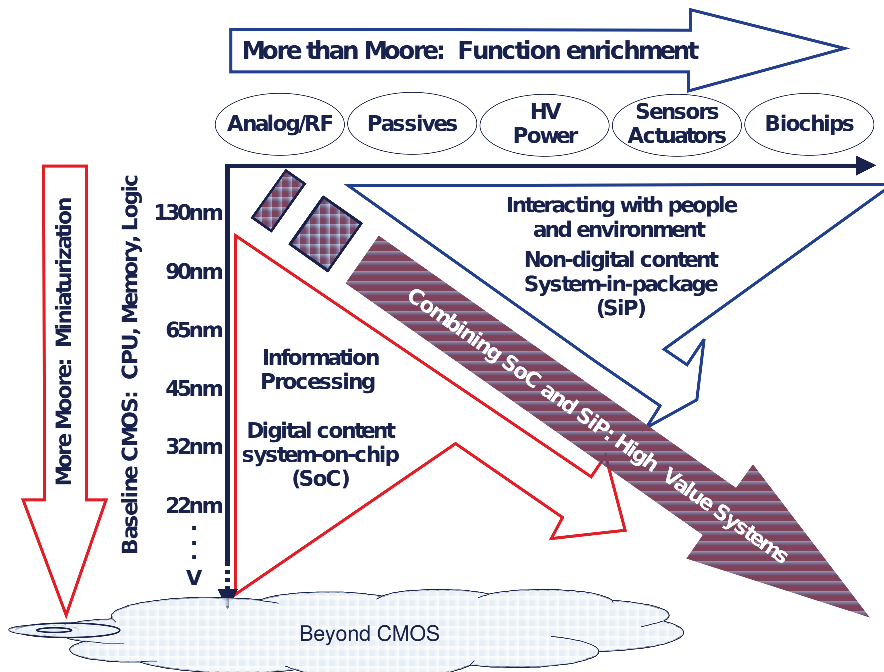
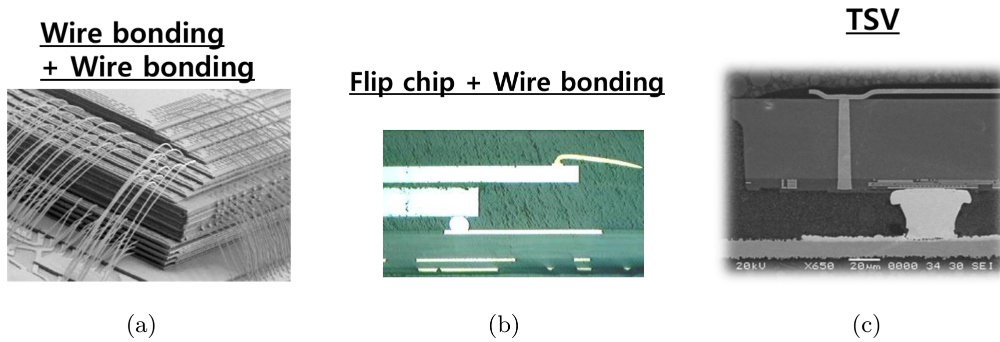


Figure 1.4: ITRS nanoelectronics technology roadmap [10].



(a) (b) (c)

Figure 1.5: Various 3D integration paradigms: (a) 3D stacking with wire-bonding, (b) 3D stacking with wire-bonding and flip-chip bonding, and (c) 3D stacking with TSV. Adapted from [11].

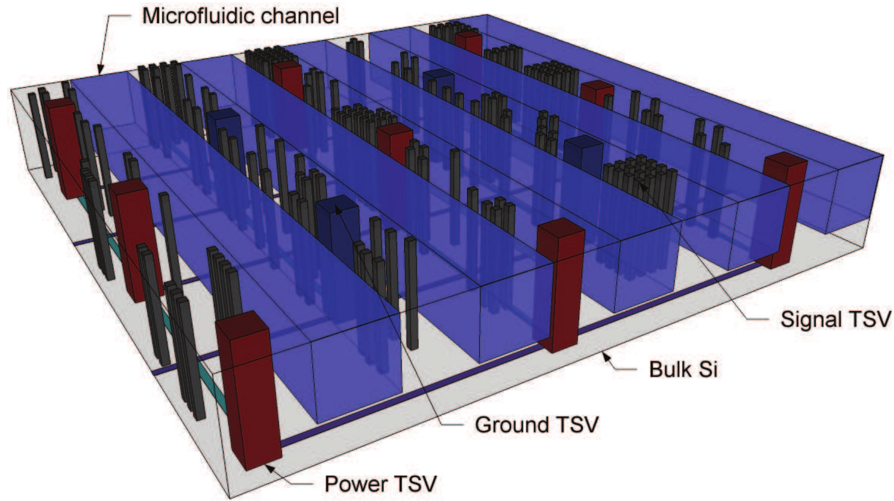


Figure 1.6: The structure of 3D IC with P/G and signal TSVs [12].

more complex functionality and improved performance. Although the advances in semiconductor fabrication technology have enabled higher transistor density by scaling down the feature size of transistors, global interconnect wire delay does not scale accordingly, and the increasing wire delays become one of the major impediments for performance improvement. As predicted by ITRS (illustrated in Figure 1.7), there is a significant performance gap between interconnect Resistive-Capacitive delay (particularly the delay caused by the long global interconnects) and gate delay, which will be increasing exponentially. As a result, long global interconnects have become a bottleneck for the overall system performance.

3D integration offers an opportunity to continue performance improvements by reducing both average interconnect length and critical path length. As the key enabling technology in 3D ICs, TSVs can provide signal paths between the tiers in a stack vertically. By employing them, the communication between electrical components packed into different tiers can be achieved by short and relatively low resistance vertical connections, instead of the long and highly resistive interconnects in the 2D counterpart. Theoretically, 3D integration can reduce interconnect length by a factor of $\sqrt{N}$ when converting a 2D chip into an N-plane 3D IC [15]. The reduction in interconnect length not only reduces delay substantially, but also leads to significant saving in power consumption due to less switching capacitance [16].

1.1.2 Memory bandwidth improvement

Nowadays, the performance of electronic systems is increasingly dominated by the time and power required to access system memory. Figure 1.8 illustrates the relative contribution of computational delay and memory access delay. As shown, compared to the 10% to 20% of the execution time spent in computation, the majority of the execution time is consumed in memory access. Furthermore, these numbers could be even more dramatic

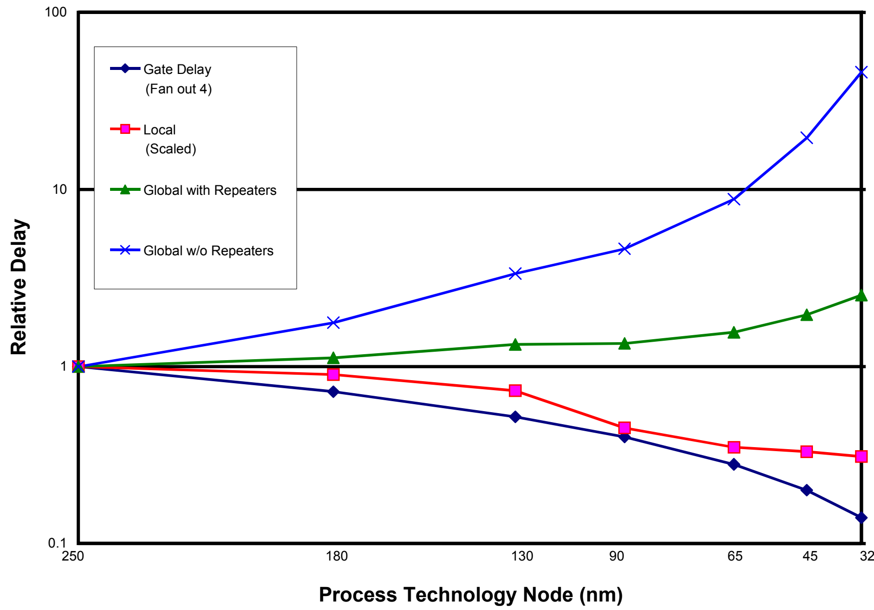


Figure 1.7: Relative delay for local and global interconnects and for logic gates in sub-micron technology nodes [14]

.

when the computation is conducted using a general-purpose processor instead of the state-of-the-art machine learning accelerator, as shown in Figure 1.8. Consequently, any radical improvement in electronic system performance has to rely entirely on a significant reduction in memory access time and power. This limitation (often referred to as the "memory bottleneck") is driven by conventional von Neumann architecture, which relies on the separation of memory and logic to facilitate computing through the operation of logic processors on separate memory stores.

The opportunity to overcome this barrier hinges on locating data closer (temporally and energetically) to processors: on the one hand, data needs to be moved quickly between processors and memories; on the other hand, only a small amount of power is consumed to transmit a bit of data between processor and memory. In this direction, a more aggressive strategy for latency reduction is to fabricate systems in three dimensions, in which the memory and logic can be integrated into a 3D stack. This powerful approach can dramatically increase the width of buses to memory through much finer-pitch vertical interconnect while simultaneously decreasing the propagation delays through shorter interconnect line. This combination can lead to memory access bandwidth as high as 40 Tb/sec for a 3D system as compared with 400 Gb/sec bandwidth in a state-of-the-art 2D counterpart [17]. In addition, another primary advantage of this approach is that it preserves the von Neumann architecture. This way, all the existing approaches to computing and investments in software tools can be reused to the utmost extent.

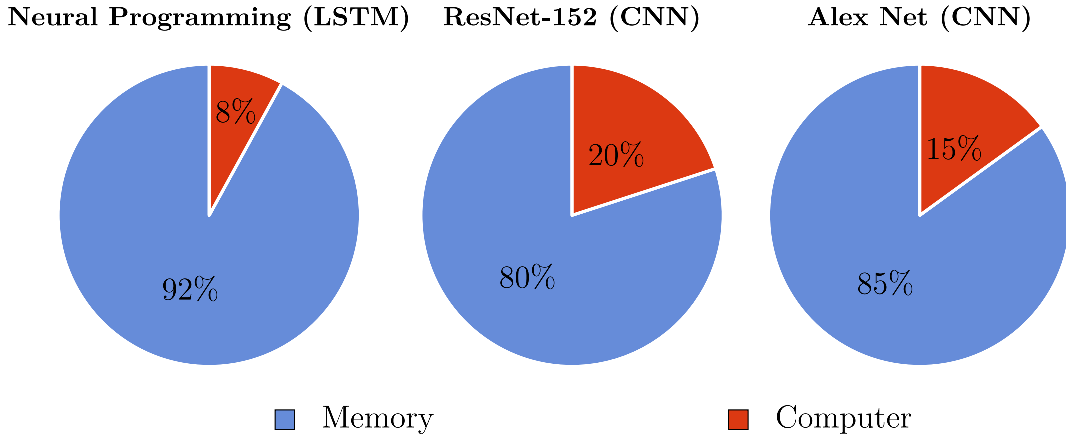


Figure 1.8: Comparison of clock cycles spent during computation and in memory access for three deep neural network algorithms as run on a machine learning accelerator executed in 7 nm CMOS technology. Reproduced from data by [17].

1.1.3 Heterogeneous integration

Nowadays, the limits of traditional Moore's law scaling in terms of technical and commercial aspects are already evident. As a result, innovation in assembly and packaging enabling functional diversification through heterogeneous integration and allowing scaling in the third dimension must pick up the slack. In particular, 3D integration provides a feasible and cost-effective approach for integrating heterogeneous components either in the sense of different functionalities, different process nodes or completely different materials. This way, different electrical components, such as analog sensors, digital logic, memory, *Radio Frequency* (RF) modules, and *Micro-Electro-Mechanical Systems* (MEMS), can be integrated together, extending the system paradigm to various new applications and feasible *System on Chip* (SoC) designs. Furthermore, by allowing different design units to be fabricated in a manufacturing process optimized for specified component, the quality of current SoC design can be improved significantly.

Leveraging 3D heterogeneous integration, it is possible to stack non-volatile memories such as *Magnetoresistive Random-Access Memory* (MRAM) or Phase-Change Random-Access Memory on top of microprocessors as on-chip memories [18, 19], enabling a new generation of processor architectures with unique features. For instances, it has been demonstrated that the optimized MRAM L2 cache can improve performance by 4.91% and reduce power by 73.5% compared with the conventional Static Random-Access Memory L2 cache [20]. In addition, with the heterogeneous integration capability offered by 3D technology, optical device layers can be stacked with CMOS logic dies in order to meet the off-chip communication bandwidth requirements at acceptable power levels. For example, HP Labs has proposed a Corona architecture [21], in which nano-photonic communication is adopted in a 3D many-core architecture for both inter-core and off-stack communications with much lower power consumption.

In conclusion, 3D integration has been considered as an enabling technology to reinvigorate Moore's Law performance scaling by: i) reducing interconnect delay, ii) breaking the "memory wall", and iii) facilitating new heterogeneous computer architectures. However, along with these opportunities and benefits, 3D integration technology introduces new challenges, which will be reviewed in the next section.

1.2 Challenges in 3D Integration Technologies

3D integration technology brings the potential of many advantages at both the circuit- and architectural-levels. However, these advantages come with a cost in terms of physical constraints and increased dependencies across metric domains. In addition, along with the performance improvement introduced by 3D ICs, new challenges have emerged, as outlined below.

1.2.1 Thermal management challenges

One of the major design challenges associated with 3D integration is thermal management, especially for homogeneous logic integration. First, due to its vertical stacking in nature, 3D integration increases transistor density without reducing the power consumption per transistor, resulting in excessive power flux as multiple tiers are stacked. Second, the dielectrics between functional tiers have relatively low thermal conductivity, which significantly diminishes heat flow from stacked tiers in a 3D IC to its heatsink in traditional air-cooling schemes and causes high temperatures. This increased chip temperature plays an important role in both soft and hard error reliabilities: high temperatures can cause timing violations, and excessively high temperature can even cause permanent physical damage to the chip. Furthermore, the chip temperature significantly effects leakage power of each transistor. As a consequence, increased power results in higher current density which can cause *Electromigration* (EM) and voltage drop in *Power Delivery Network* (PDN).

Moreover, the heat generation from a 3D IC is non-uniform spatially, especially with the areas of very high local heat fluxes at few locations on a die. Consequently, an increase in thermal design power can lead to an increase in both average power density and local power density (often referred to as "hot spots"). In addition, even when the total power consumption of the 3D stack remains within design specification, the local power density could also be increased, which limit the performance and reliability of 3D system. For example, temperature fluctuations can lead to TSV defect formation from thermal cycling (i.e., so called TSV pop-out and delamination [22]). For a 3D IC, thermal management must take account of the hot spot thermal dissipation within the stack as well as within the package. As a result, it is imperative to model and analyze the thermal behaviors of 3D ICs accurately. Furthermore, sophisticated thermal-aware design techniques should be developed for 3D ICs with a holistic approach from circuit-level up to system-level.

1.2.2 3D power delivery network design challenges

As the benefits of Dennard scaling [2] quickly vanish recently, it becomes increasingly challenging to keep the historical trend of device integration with the conventional planar fabrication approach. The vertical 3D integration technology provides an alternative pathway toward continuing the exponential growth in device integration, but also increases the aggregated density of both power consumption and heat generation during operations. Consequently, it is becoming extremely challenging to deliver sufficient current to each circuit module as well as to dissipate the massive heat generated in a 3D stack. On the one hand, the upper-level tiers suffer from more severe voltage noise since they are further away from power supply pads [23]. On the other hand, the lower-level tiers experience higher temperature as the heatsink is attached to the top of the 3D stack with conventional cooling system. As the number of tiers in a 3D system is expected to increase in the future, the problems of delivering power supply voltage to each transistor without excessive power noise along the power delivery path become more daunting. In order to alleviate the power delivery constraints in the era of 3D ICs, PDN must be carefully designed for maintaining system-wide power integrity, which is believed to be one of the biggest challenges.

PDN deliver power and ground voltage from package to the devices. In a 3D IC, the individual PDN on each tier has to be interconnected by the vertical connections such as TSVs (as shown in Figure 1.9), and thus the challenges for a reliable 3D PDN design are multi-fold. First, due to its intrinsic resistance, capacitance, and inductance, the supply voltage will become noisy (either drop or fluctuate) after travelling through the PDN. As the propagate delay of each logic gate is highly dependent on the supply voltage, any voltage variation exceeding the pre-defined design margin can cause a timing error, threatening system performance and program correctness. This power supply noise can be further exacerbated by the higher parasitic impedance of 3D PDNs. Second, the 3D PDN can suffer from long-term reliability issues such as EM. Under the higher current stress in 3D ICs, EM can cause permanent damage to the metal interconnects and TSVs in a PDN, leading to the degradation of power delivery quality with increased power supply noise and the associated soft errors. Third, due to the heterogeneous integrations in 3D ICs, certain low power design techniques (such as power gating) are often implemented, which further complicates the PDN due to the individual control of different power domains in the system. To deal with these challenges, novel and sophisticated design methodologies considering the unique characteristics of 3D PDNs must be developed. In particular, considering the fact that TSVs are the only channels for power-supply connections in a 3D stack, the allocation of TSVs can introduce a trade-off between power delivery quality and area overhead, and should be investigated thoughtfully.

[2] Dennard scaling, also known as MOSFET scaling, is a scaling law based on a 1974 paper co-authored by Robert H. Dennard. Roughly speaking, it states that as transistors get smaller their power density stays constant, so that the power use stays in proportion with area: both voltage and current scale (downward) with length.

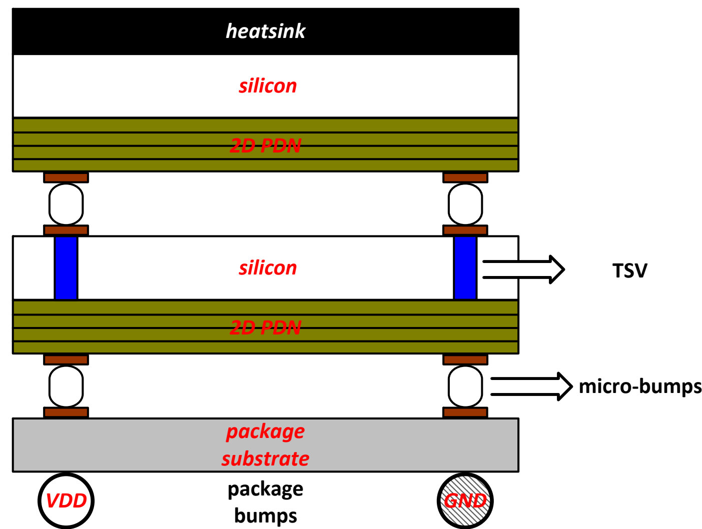


Figure 1.9: Two-tier 3D IC using face-to-back TSVs

.

1.2.3 TSV-related reliability and yield challenges

Although 3D chips will experience the same environmental, mechanical, and electrical stresses as in 2D chips during operations, the extra silicon components (such as TSVs) can add more structures and complexity to the system. This additional level of electrical interconnections introduces new failure mechanisms for the 3D system. Various manufacturing and latent TSV defects arise from the disruptive manufacturing process involved during TSV fabrication, and aggravated by the coupling effects between electrical, mechanical, and thermal conditions. In particular, the TSVs and back-grinding of the silicon wafer can cause significant thermo-mechanical stresses, which not only result in systematic mobility/performance variations, but also lead to mechanical reliability concerns (e.g., interfacial cracking) [24]. Additionally, coupling of TSVs to substrate [25, 26] and among each other [27, 28] may give rise to severe power and signal integrity issues, which can be further aggravated by the capacitive and inductive coupling between two adjacent tiers [29, 30]. Moreover, reliability and yield challenges might increase overall cost [31–33]. The benefits provided by 3D integration can be realized only when the design and manufacturing cost of products is commercially viable [34]. Among all the factors affecting the cost of 3D ICs, manufacturing yield is one of the most crucial ones. It has been shown that the functional yield of a rather simple three-tier chip is only a bit more than 60% [35]. Therefore, it is very critical to identify, evaluate and understand the reliability and yield challenges associated with TSVs in 3D integration.

1.2.4 Testing challenges

Adding to these design challenges in 3D ICs is the increased complexity of testing. A wide adoption of 3D integration is hampered by an insufficient understanding of 3D

testing issues and by the lack of design for *Design-for-Testability* (DfT) techniques. Due to the unique processing steps of 3D integration (such as wafer thinning, alignment, and bonding), a vast number of new defect mechanism arise in 3D ICs, and must be addressed as part of a test strategy. However, one of the major bottlenecks for 3D testing is the loss of access to package pins which are often re-purposed as scan I/Os. As a result, the number of test pins available for testing a die in a 3D IC is far limited, which is much smaller than that in 2D package. Without full access to the logics on each die, test coverage could be reduced and the possibility of defects escaping to the stack increases, risking product quality and manufacturing yield. Furthermore, since target market segments for 3D integration (including CMOS image sensor, 3D memory, etc.) are mainly in consumer electronics at present, any product for this price-sensitive market with excessive test cost will not be successful. Here, the major elements of test cost are test-application time, test power consumption, and cost of test equipment, etc. As a result, proper DfT and testing schemes is imperative to maintain test quality with lower test costs.

The introduction of TSVs in 3D ICs also raises many testing challenges. In order to relieve the impact of TSV-induced thermal stress on circuit performance, a Design-for-Manufacturing and/or Design-for-Reliability methodology needs to be implemented. Moreover, for the improvement of the yield and reliability of TSVs, *Built-In Self-Test* (BIST) redundant structures and in-field repair mechanisms are imperative.

The main objective of this dissertation is to tackle the aforementioned technology challenges in 3D integration. To this purpose, a set of comprehensive solutions were developed, including *Computer-Aided Design* (CAD) methodologies and optimization techniques for reliability, timing performance, and testability of 3D ICs. Below is a summary of the main contributions and outline of this dissertation in the various topics of research.

1.3 Dissertation Contributions and Outline

The main contributions of this dissertation is to tackle the aforementioned technology challenges in 3D integration. To this purpose, a set of comprehensive solutions were developed, including CAD methodologies and optimization techniques for reliability, timing performance, and testability of 3D ICs (as shown in Figure 1.10). First, a holistic CAD platform is proposed to address the need for reliable PDN design in a 3D integration system in terms of optimal P/G TSV planning from power and signal integrities perspective (i.e., Contribution I in Figure 1.10). Afterwards, a comprehensive framework of signal TSV repair is developed for yield enhancement and reliability improvement in 3D ICs (i.e., Contribution II in Figure 1.10). Last, a multicast test approach is proposed for interposer-based 3D ICs in order to reduce test costs (in terms of test-application time and test power consumption) and maintain high test quality (i.e., high fault coverage). (i.e., Contribution III in Figure 1.10).

The details of our contributions in this dissertation are summarized as follows.

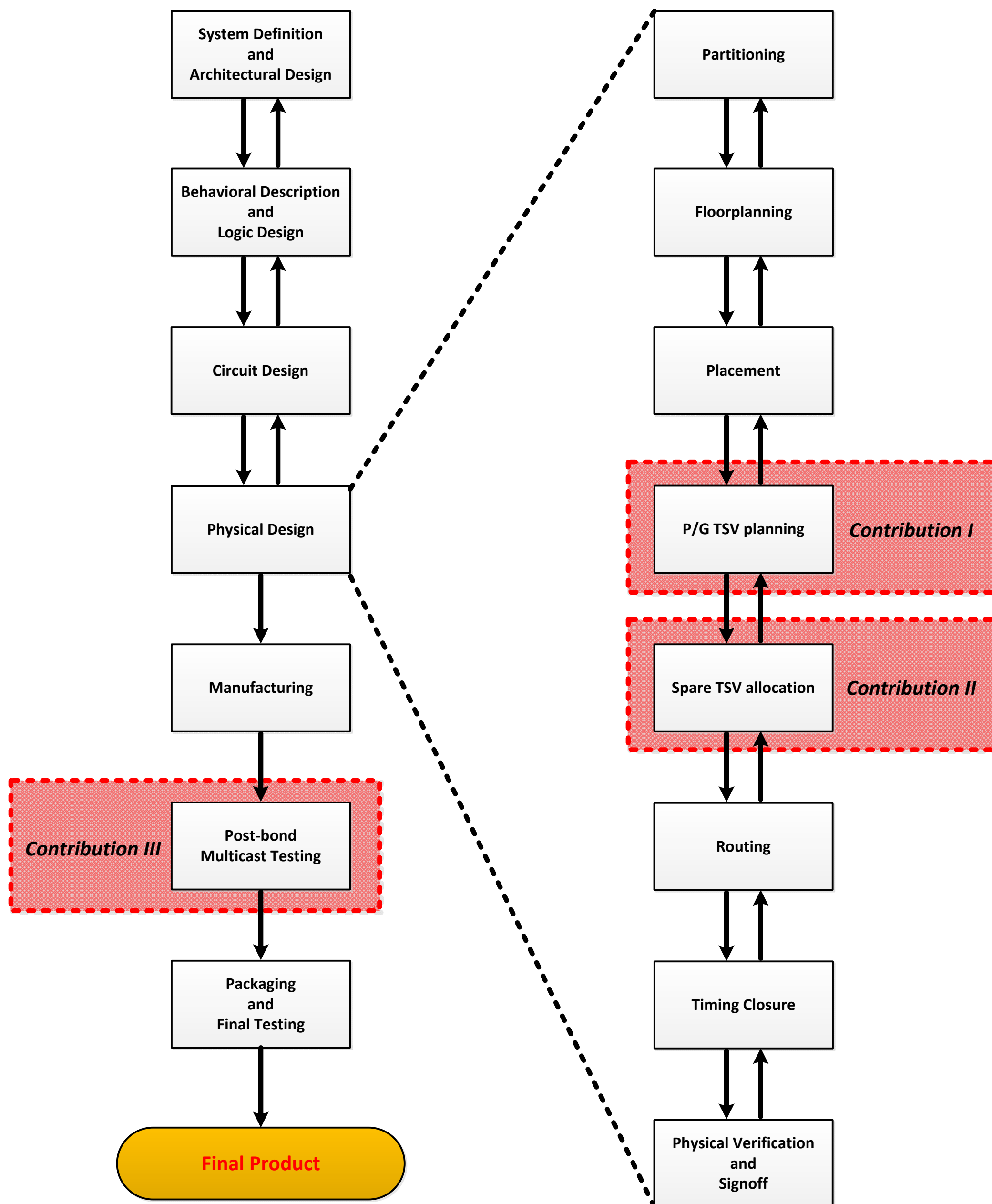


Figure 1.10: Proposed reliability-aware physical design and test flow for 3D ICs in this dissertation.

1.3.1 Contributions to reliable 3D power delivery network design

Thermal stress-aware non-regular power/ground TSV planning technique

In 3D ICs, smaller footprints combined with multiple device layers imply severe power delivery problem, which is further exacerbated by the limited number of P/G TSVs. Compared to other layout objects in a chip, TSVs are quite huge in size, and thus become significant obstacles during placement and routing. In particular, P/G TSVs occupy either bulk and device layers in case of via-first process or entire 3D stack in case of via-last process. Consequently, the number of TSVs available for P/G nets is limited under the constraints of design flexibility, making a reliable 3D PDN design highly challenging. In this dissertation, a non-regular P/G TSV planning method based on *Integer Linear Programming* (ILP) formulation is proposed to minimize power supply noise while reducing the area overhead introduced by P/G TSVs to an acceptable level. The results from the conventional regular and the proposed non-regular TSV planning approach are compared in terms of IR-drop noise and P/G TSV count. Furthermore, the impact of TSV-induced thermo-mechanical stress on 3D PDN robustness is studied, in which both voltage drop- and thermal stress-induced delay variations are taken into account during P/G TSV planning. As the important metrics of risk in 3D PDN design, the impact of *Keep-out Zone* (KOZ) of TSV and PDN mesh granularity is also investigated.

Electromigration-aware local-via allocation approach in power/ground TSVs

There exist three different types of TSVs dictated by the manufacturing process: via-first, via-middle, and via-last TSVs. In the via-first TSVs, local-via, which is used to connect landing pad of TSV and 2D PDN, plays an important role for balancing the trade-off between EM reliability and design routability. On the one hand, in order to improve the EM-related lifetime reliability of P/G TSVs, it is desirable to insert multiple local vias inside each TSV for load balancing of the current density per via, which can increase EM-related lifetime effectively. On the other hand, excessive local vias for each P/G TSV can consume too much routing area, leading to exacerbated routing congestion and increased delay overhead. In this dissertation, a routability-aware local-via allocation approach is explored to minimize the routing congestion introduced by local vias while satisfying the constraints of EM-induced power supply noise during field operations. To the best of our knowledge, this is the first work that takes design routability into account during local-via allocation. In addition, the impacts of temperature, 3D design partitioning, and expected TSV lifetime on local-via allocation are also investigated.

1.3.2 Contributions to TSV self-repair of 3D ICs

TSV self-repair framework for assembly yield enhancement

In order for 3D ICs to be commercially viable, a set of yield enhancement solutions through self-repair is required. In general, the yield of 3D ICs can be reduced due to the

defects in stacked dies (i.e., stack yield loss) or defects that occur during the assembly process (i.e., assembly yield loss). Therefore, the yield of a 3D IC is even less than that of each individual die/wafer in the 3D stack. For stack yield enhancement, it is critical to conduct pre-bond testing to prevent the stacking of defective dies. For assembly yield enhancement, the addition of spare TSVs to repair defective ones is the most effective method besides improving the manufacturing process itself. In this dissertation, a novel TSV self-repair framework is proposed with the consideration of defect clustering effect, which is based on a light-weight hardware architecture associated with efficient algorithms for spare TSV allocation. In the previous TSV repair schemes, there are two major drawbacks: i) the claimed yield enhancement cannot be guaranteed under realistic clustering defect distribution; ii) since the delay variation introduced by thermal stress was ignored, the requirement of timing correctness for the system would be violated after TSV repair. The proposed solution in this dissertation overcomes the above limitations by considering the trade-off between chip yield, hardware cost, and delay overhead.

TSV self-repair framework for run-time reliability improvement

TSV fabrication involves disruptive manufacturing technologies, and has been shown to introduce new failure mechanism in 3D ICs. In particular, the huge dimensional gaps between TSVs, on-chip metal wire, and bonding can lead to new EM concerns compared to conventional 2D planer interconnects, which pose reliability threats for TSVs and cause hard-to-predict timing errors during field operations. In this dissertation, a proactive repair approach is proposed to combat TSV EM degradation by taking use of the transient recovery effect in EM-induced stress evolution. In conventional repair solutions, this recovery effect is ignored, and thus TSV redundancy is used in a reactive manner, in which the operability of a 3D chip in the presence of EM failure is maintained by detecting and replacing faulty TSVs with spares. This way, EM-related reliability improvement is restricted to the amount of pre-allocated spare TSVs, which is normally limited due to its associated area overhead. By contrast, the proposed approach can take full advantage of the TSV redundancy resource by exploiting the EM recovery effect, and extend EM-related lifetime of multiple TSVs even with one single spare.

1.3.3 Contributions to cost-effective testing of 3D ICs

Multicast testing of interposer-based 3D ICs

As a precursor to TSV-based 3D ICs, interposer-based 3D ICs (often referred to as *Two-and-a-Half-Dimensional Integrated Circuits* (2.5D ICs)) have gained increasing attention recently. However, it is still a major challenge to test 2.5D ICs using conventional methods due to the limited number of dedicated test pins in the package level. In addition, because of higher integration levels in 2.5D ICs, test-application time and test power are also increased compared to their 2D counterparts, resulting in time-consuming and costly testing solution. To tackle these issues, a cost-effective multicast test approach is proposed to target die testing in 2.5D ICs at post-bonding stage. Its main idea is to

apply the same test patterns to carefully selected multiple dies when they can effectively achieve high fault coverage for these dies within a given test-power budget. This way, the number of required test pins is significantly reduced, and thus die testing of 2.5D ICs can be accomplished using *Automatic Test Pattern Generation* (ATPG) patterns with a limited number of test pins. To this end, an efficient test architecture is developed with negligible area overhead. Meanwhile, a test-scheduling and optimization technique for identifying groups of dies for multicast is also proposed.

1.3.4 Dissertation outline

The remainder of this dissertation can be mainly divided into four parts. First, background information and state-of-the-art techniques about the 3D integration technology are presented in Chapter 2. Next, Chapters 3 and 4 cover the proposed reliable 3D PDN design approaches, consisting of thermal stress-aware P/G TSV planning and local-via allocation in P/G TSV for EM reliability improvement. The third part, including Chapters 5 and 6, focuses on the design of TSV repair solutions for yield enhancement and reliability improvement. In addition, the cost-effective multicast testing approach for 2.5D ICs is described in Chapter 7. Finally, Chapter 8 concludes the dissertation.

2 Background

In this chapter, a general description about interconnect hierarchy in 3D ICs is presented in Section 2.1. Next, the existing applications and future opportunities of 3D integration technology are overviewed in Section 2.2. Then, the big-picture of TSVs is presented. As the key and most popular enabler of 3D ICs at present, TSV and its reliability are the major research topic in this dissertation. The different fabrication approaches of TSVs are introduced in Section 2.3. Afterwards, the major reliability issues of TSVs, including thermo-mechanical stress and EM, are briefly explained in Section 2.4. Last, the basic flow of 3D ICs is reviewed in Section 2.5.

2.1 Interconnect Hierarchy in 3D Integration

Nowadays, most electronic system are characterized by a strong hierarchical interconnect structure, as shown in Figure 2.1. At transistor level, the interconnects are narrow and short (i.e., local interconnect). In order to connect blocks of transistors, longer lines with larger cross-sections are used (i.e., intermediate interconnect), while large circuit blocks (e.g., *Intellectual Property* (IP) blocks) are integrated with the longest and widest wires (i.e., global interconnect). The main driving force of such multi-layer interconnection is to: i) achieve higher packing densities and smaller footprint, and ii) mitigate resistive-capacitive delay. In order to route the same signal between different interconnect planes, connections between adjacent interconnect planes are realized through "vias" interconnects (e.g., damascene contacts, plated through hole, pins, solder balls, connectors, etc.). For the interconnections between different ICs, the electronic signals can be buffered and connected to a bond-pad interconnect level, of which the pitch is larger to allow for compatibility with the package-level interconnect capabilities. This type of hierarchy of interconnect levels is continued in the package and *Print Circuit Board* (PCB) interconnect layers, in which various structures are used, such as wire bonding, solder bumps, metal pillars, and *Redistribution Layer* (RDL).

In particular, for a 3D system, although several interconnects such as flexible printed circuit boards and molded interconnect devices can be shaped or folded in three dimension, their topology still remains essentially two-dimensional. The 3D system integration requires the capability to make direct connections in the third dimension, connecting different layers of active devices that are connected by a 2D-interconnect fabric. To this purpose, the most popular solution is a special type of "via" interconnect, which penetrates through an entire silicon wafer, known as TSV. On the other hand, the 3D connections can also be realized at the board or package level.

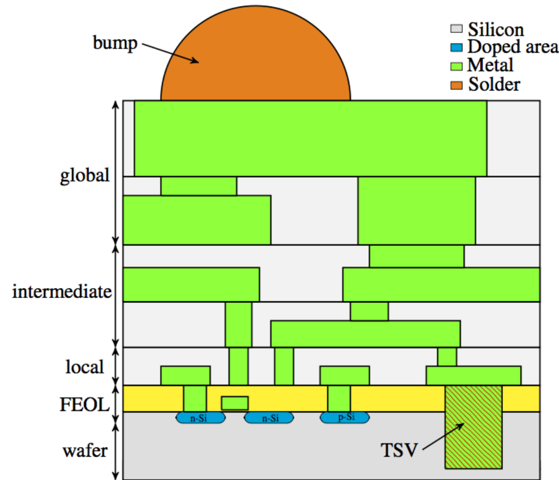


Figure 2.1: Classification of interconnects in a 3D system [36].

2.2 3D Integration Technology Landscape and Applications

3D integration has been very active over the past ten years, and pursued by almost all players in the electronic manufacturing supply chain (such as manufacturing foundry, wafer level processing, semiconductor assembly and test, PCB, etc.). As a result, a broad variety of technologies has been proposed, and the traditional interfaces between them are blurring. In addition, different specific applications (such as high-density memory, high voltage, analog signalling, and sensors, etc.) also drive 3D integration technology developments in various directions. In order to come to a clear vision on roadmaps for 3D integration technologies, it is imperative to provide clear definition and classification on the wide variety of technologies that capture the functional requirements of 3D technology at the different hierarchical levels of the system and correspond to the supply chain manufacturing capabilities.

In this complex landscape of 3D technology, several classes of 3D integration can be defined based on the level of partitioning (as shown in Figure 2.2), meaning that the system is divided into different pieces at various levels in the interconnect hierarchy. Each of these classes requires different process schemes and 3D integration techniques.

- The first class is 3D *System-in-Package* (SiP), in which the partitioning is done at package level, either by stacking packages on top of each other or by integrating multiple die in a single package. Among the technologies used to integrate SiP are *Package-on-Package* (PoP) and *Fan-out Wafer-Level Packaging* (FO-WLP), in combination with traditional interconnect technologies, such as wire bonding, flip chip, and solder bumps [37].

- The second class is 3D *Stacked Integrated Circuit* (SIC), in which the partitioning is done at die (and/or block) level. In a 3D SIC, multiple dies are vertically stacked

using TSVs and micro-bumps. Stacking approaches of ICs are *Die-to-Die* (D2D), *Die-to-Wafer* (D2W), *Wafer-to-Wafer* (W2W), and Die-to-Interposer.

- The third class is 3D *Monolithic Integrated Circuit* (MIC), in which the partitioning is done at the level of standard cell (or even transistor). The integration scheme for a 3D IC is achieved by direct stacking of device layers using connections at the density level of local interconnects, instead of a TSV-like processes. In detail, the interconnect will be realized through sequential processes or layer transfer processes, where stacked 2D device layers are connected together vertically on a single wafer. Consequently, the 3D system obtained is treated as a monolithic structure.

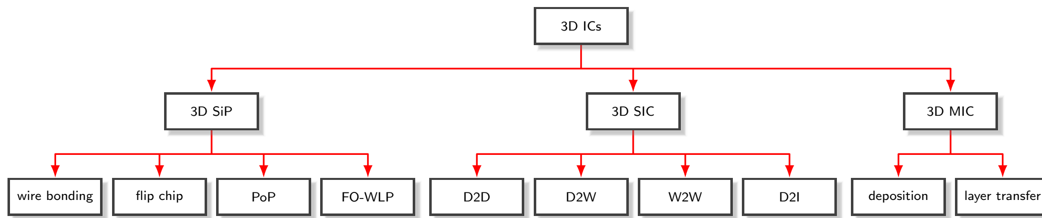


Figure 2.2: Classification of 3D integration technologies.

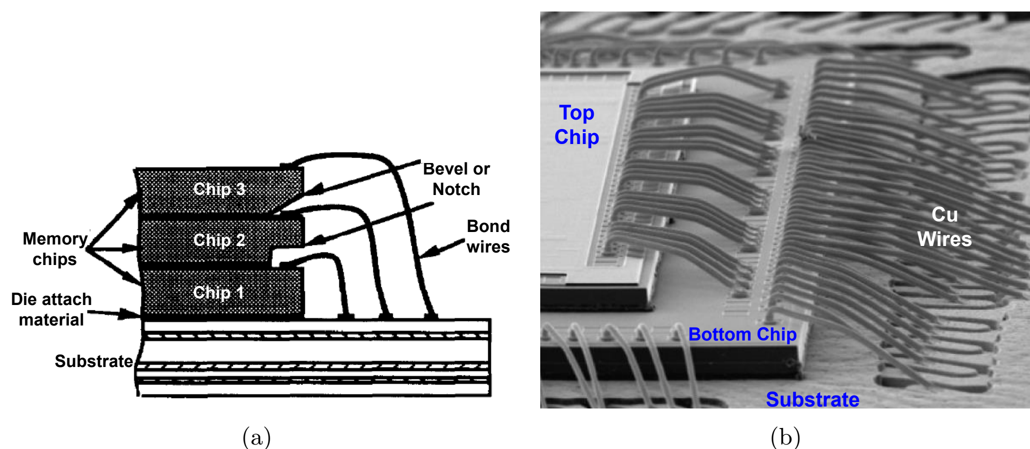
2.2.1 3D system-in-package

Over the last decades, many technologies for 3D SiP have been proposed and brought to *High-Volume Manufacturing* (HVM) based on the modifications of established packaging technology platforms, which are typically characterized by short development times, low fabrication cost, and high yield. As a result, 3D SiP can offer the highest versatility and typically lowest costs of the system, helping bring products to market quicker when technical and/or business reasons prevent timely SoC integration. Among all the 3D SiP configurations, chip stacking by wire bonding and PoP are now mature for HVM, and 3D FO-WLP is also a potential candidate for production.

Bare chip stacking by wire-bonding

The most widespread 3D SiP technology is probably the stacking of bare chips in a package using wire bond connection to the package substrate or lead frame, and its popular application areas include memory die stacking and sensor die on Application-Specific Integrated Circuit chip stacking. The first paper on this configuration was published by nCHIP more than 20 years ago [38], in which multiple memory chips are stacked vertically by die-attach material and gold wire bonding (as shown in Figure 2.3(a)). Since then, 3D memory stacking by wire-bonding has been in high volume production for the applications of smart-phones and tablets (e.g., Samsung's eight-stack flash in Apple's

iPhone 4s and SK Hynix's multi-level-cell 8GB NAND flash in Apple's iPhone 5s [37]). Moreover, due to the surge in gold prices, many companies have been pursing low-cost solutions, and the shift from gold to copper wire-bonding technology is genuinely picking up (e.g., AMKOR's 3D IC packaging shown in Figure 2.3(b)).



(a) (b)

Figure 2.3: (a) nCHIP's 3D memory stacked using gold wire-bonding [38] (b) AMKOR's 3D IC packaging stacked by copper wire-bonding [37].

Packaged chip stacking by package-on-package

PoP is perhaps the most important 3D SiP configuration today in the smart-phone and tablet market, which refers to the 3D integration scheme in which two or more packaged chips are stacked vertically with a standard interface to route signals between them [39]. At present, two widely-used configurations exist for PoP:

- Pure memory stacking: two or more memory-only packages are stacked on each other.
- Logic-memory stacking: logic package on the bottom, memory package on top.

Two examples are illustrated in Figure 2.4, in which the application processor is stacked with external Low-Power Double Data Rate DRAM as a 3D SiP for the application of mobile devices. The key advantage of this approach is not only related to the reasons for technical or performance, but also business. For building a very high-volume end application, PoP can provide a clear supply chain of all involved companies, which includes the logic device foundry, the memory suppliers, the chip packaging companies, the Outsourced Semiconductor Assembly and Test companies, and the Original Equipment Manufacturer.

Embedded 3D fan-out wafer-level packaging

FO-WLP is an enhancement of standard Wafer-Level Packaging for a greater num-

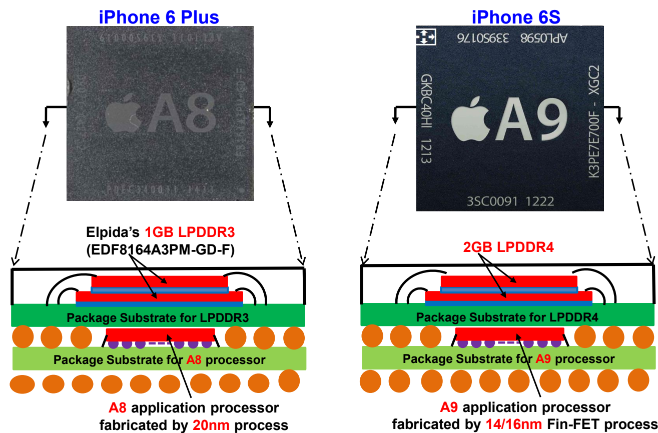


Figure 2.4: Top-view and cross-sectional view of the PoP for mobile DRAM and application processor [40].

ber of external I/Os. After dicing chips on a silicon wafer, all the known-good chips are precisely positioned on a thin "reconstituted" or carrier wafer/panel, which is then molded, following by making redistribution layer atop the molded area (chip and fan-out area), and then forming solder balls on top. The first appearance of this configuration was proposed by Infineon in fall 2007 with the *Embedded Wafer Level Ball Grid Array* (eWLB) and "Molded Reconfigured 200 mm Wafer" (as shown in Figure 2.5). This chip is a wireless baseband SoC with multiple integrated functions, such as GPS, FM radio, etc.. In early 2009, Infineon was also the first company to commercialize its own eWLB packaging technology in an LG cell-phone, and also in some Nokia handsets since 2010.

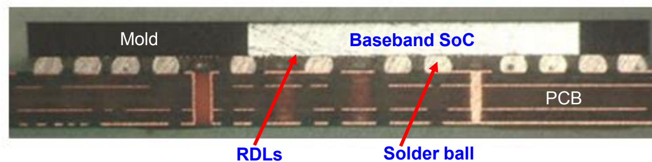


Figure 2.5: Cross-sectional view of the eWLB from Infineon [40].

2.2.2 3D stacked integrated circuit

3D SiP is always defined by the interconnect density capabilities of the package-level interconnect technology, not by the on-chip patterning and connection capabilities. In contrast, 3D SIC technology can overcome this bottleneck introduced by package-level processes, allowing for the use of much smaller connections, such as micro-bumps, TSV, and via contactless connection based on capacitive [41, 42] or inductive coupling [43, 44]. Micro-bump is nothing but a small ball of solder or other metals on the surface of the die connecting other dies together, which can provide higher interconnect density. However, they do not reduce parasitic capacitances due to the need to route signals

to the periphery of the 3D stack to each destinations within it. On the other hand, although involving few processing steps, the manufacturing difficulties and insufficient interconnect densities of the contactless approach limit its wide adoption. As a result, TSVs are the most promising and popular one among all these various interconnection scheme as they can provide the highest interconnect density. though they also require more manufacturing steps [45].

Stacking and bonding methods for 3D integration can be mainly classified into three categories: i) D2D, ii) W2W, and iii) D2W [46], as shown in Figure 2.6.

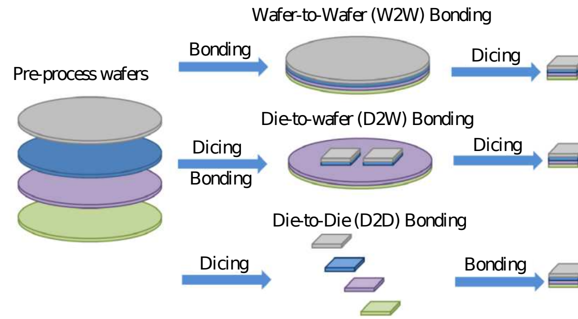


Figure 2.6: Classification of 3D integration according to different stacking approaches [47].

- In D2D stacking, electronic components are built on multiple dies, which are then aligned and bonded together. Thinning and TSV creation can be done before or after bonding. As it allow for testing of each die prior to being integrated to a 3D stack, the production yield of the D2D approach is pretty high. Moreover, since each die in the 3D SIC can be binned beforehand, they can be mixed and matched to optimize power consumption and performance individually. However, the production throughput of this approach is extremely low due to the one-by-one processes at the chip level.

- In W2W stacking, multiple wafers consisting of many copies of the same circuitry are stacked on top of one another. The stacked wafers are aligned, bonded, thinned, and diced into individual 3D SIC. Due to the batch processes at the wafer level, the production throughput in the W2W approach is very high. On the other hand, its total production yield exponentially decreases with an increase in the number of stacked layers because defective dies can not be removed from wafers to be stacked. Thus, the W2W integration scheme is more desirable for high-yield products such as DRAM.

- In D2W stacking, two wafer are produced but only one of them is diced into individual dies, which are stacked on the other wafer afterwards. Since Known-Good-Dies can be selectively used and wafer level processes can be applied after chip assembly, the 3D integration based on D2W stacking can dramatically improve the production yield. Furthermore, the traditional robotic pick-and-place chip

assembly approach can be replaced by self-assembly method [48], which leads to the increase of production throughput. The D2W integration scheme is preferred in configurations that require three or more dies in a stack.

A special class of 3D SIC is the 2.5D SIC, in which two or more active dies are placed side by side on a large passive silicon interposer providing the connection by means of TSVs and wires. Although most interposer do not contain active elements, this 2.5D integration is also considered as a 3D SIC technology. Furthermore, a possibility to integrate active elements inside the interposer is also explored [49].

3D SIC stacks up thin IC chips in the third dimension with TSVs and micro-bumps to achieve performance improvement, low power consumption, high bandwidth, and small form factor. The ones which are in and going into low volume production are: i) memory stacking with TSVs, ii) *Hybrid Memory Cube* (HMC), iii) wide I/O DRAM, iv) *High Bandwidth Memory* (HBM), and v) passive interposer.

Memory chip stacking

In August 2014, Samsung mass-produced the first industry-level TSV-based 64 GB *Double Data Rate* (DDR)4 DRAM module for server applications (as shown in Figure 2.7), which consists of 36 DDR4 DRAM chips [50]. In each chip, four 4 GB DDR4 DRAM dies are stacked together using Samsung's 20 nm process technology and 3D TSV packaging technology. The module performs twice as fast as a module that uses wire bonding packaging with approximately 50% power saving.

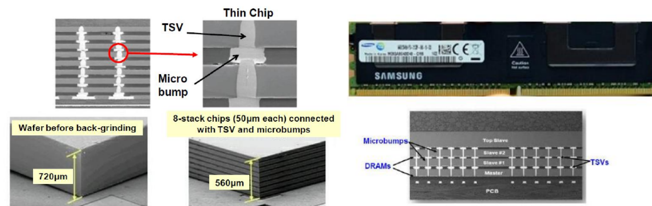


Figure 2.7: Samsung's 64 GB DDR4 DRAM module for server farm [40].

Hybrid memory cube

HMC is a high-performance *Random-Access Memory* (RAM) interface for TSV-based stacked DRAM memory, which was announced by Micron Technology in 2011 [51]. HMC uses standard DRAM cells but it has more data banks than classic DRAM memory of the same size, and thus promises a higher speed improvement over DDRn (around 15 times compared to DDR3 [52]). The first sample shipped by Micron/IBM at the end of September 2013 is shown in Figure 2.8, which consists of four DRAMs with more than 2,000 TSVs individually stacking on top of a logic controller [53]. Afterwards, the HMC is attached to an organic package substrate. The TSV-based DRAM cube was fabricated by Micron and the controller for TSVs was by IBM.

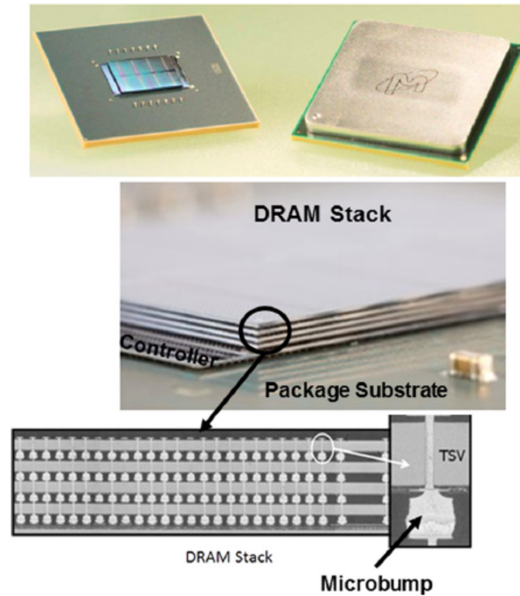


Figure 2.8: Micron's sample on HMC [53].

The first processor to use HMCs was the Fujitsu SPARC64 XIfx, which is used in the Fujitsu PRIMEHPC FX100 supercomputer introduced in 2015 [54]. In addition, Intel's Knights Landing processor unit support for up to 384 GB of on-board DDR4 RAM and 16 GB of on-package HMC stacked DRAM, which provides up to 500 GB/s of memory bandwidth [55]. According to the reports from Micron, having such HMC in the package of this processor is expected to deliver 5× the sustained memory bandwidth compared to *Graphics Double Data Rate* (GDDR)5 with around 30% power consumption per bit in half the footprint.

Wide I/O DRAM

Wide I/O and Wide I/O 2 have been backed by companies (e.g., Samsung) and are designed to provide mobile SoCs with a maximum amount of bandwidth at the lowest possible power consumption. They are meant for a stack of DRAMs with TSVs and the associated TSV controllers on top of SoCs, which is very similar of the HMC. Although this approach optimizes the size of package, but also imposes severe thermal limitations as heat radiated from the SoC has to pass through the entire memory stack. As a result, operating frequency has to be lower, but a larger number of I/O pins increases bandwidth by using a memory bus that's up to 1,024 bit wide. Explicitly, Wide I/O can be designed as a 3D interface. However, in order to cool the CPU underneath the stacked DRAM chips, 2.5D interposer designs are more desirable in this scenario.

High bandwidth memory

HBM is a high-performance RAM interface for 3D-stacked DRAM proposed by AMD and Hynix, which is to be used in conjunction with graphics accelerators and network devices. The first device to use HBM is the AMD Fiji *Graphics Processing Units* (GPUs) [56]. Compared to DDR4 or GDDR5, HBM can achieve higher bandwidth with less energy in a substantially smaller form factor. This is realized by stacking up to eight DRAM dies, including an optional base die with a memory controller. The interconnections between dies are provided by TSVs and micro-bumps. As shown in Figure 2.9, a TSV/RDL interposer is used to support/connect the lateral communication between the HBM DRAM memory cube with TSVs and the SoC (such as GPU or CPU) without TSVs. The optional base chip is used for buffering and signal re-routing of the HBM DRAM cube. In principle, the HBM technology is similar but not compatible with the HMC interface.

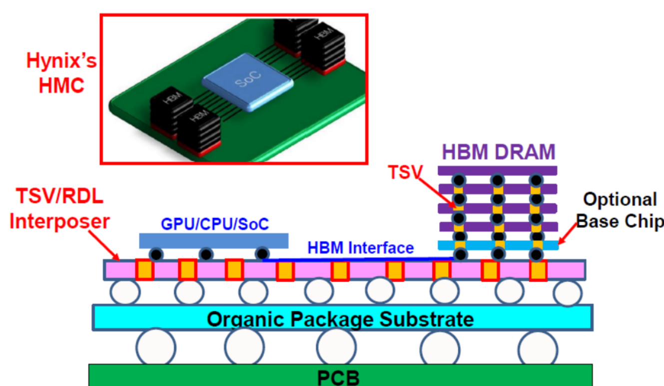


Figure 2.9: Schematic of Hynix's HBM.

Passive interposer & 2.5D integration

A 2.5D SIC is a TSV/RDL interposer-based system consisting of a piece of device-less silicon with TSVs (also called passive interposer), RDLs, and IC chips without TSVs. The passive interposer can be used to support the high-performance, high-density, fine-pitch chips, and RDLs mainly provide lateral communications between the chips, as shown in Figure 2.10. This integration scheme is in small volume manufacturing for the sliced *Field-Programmable Gate Array* (FPGA). Figure 2.11 provides a sample which is designed and fabricated by Altera/TSMC [57, 58]. It can be seen that even with more than 12 build-up layers on the package substrate, it is still not enough to support the four sliced 28 nm FPGA chips. Moreover, a passive interposer consisting of more than 200,000 micro-bumps with 45 μm pitch and four RDLs with 0.4 μm pitch is required. This so-called "*Chip on Wafer on Substrate* (CoWoS)" 3D process proposed by TSMC has also been used for the small production in Xilinx since the early of 2013 [59]

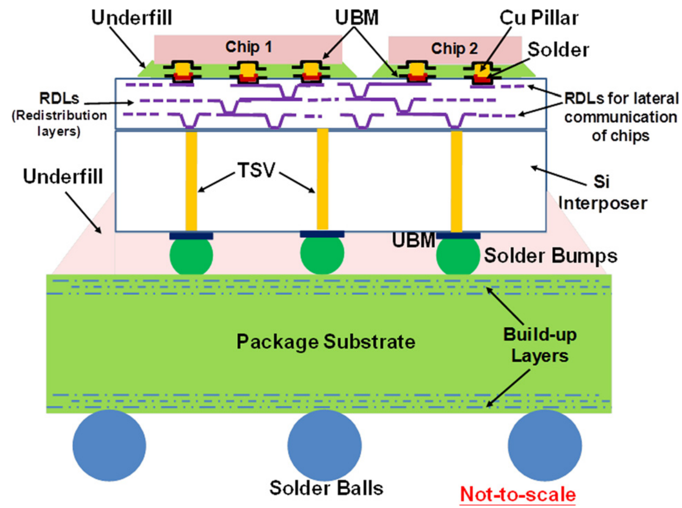


Figure 2.10: A passive interposer which consists of TSV and RDL supporting chips on package substrate [37].

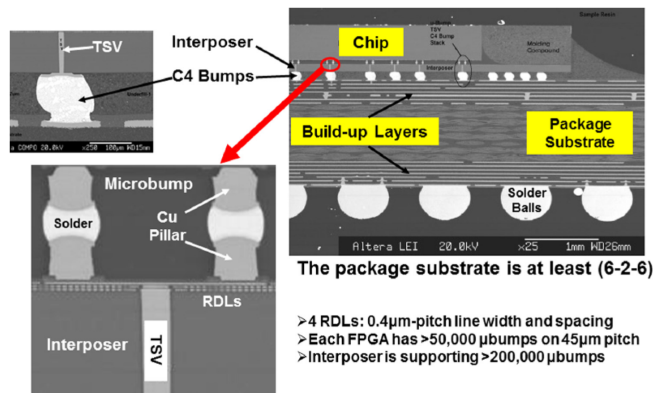


Figure 2.11: Altera/TSMC's CoWoS [58].

2.2.3 3D monolithic integrated circuit

In a 3D MIC, the device layers are fabricated sequentially, one on top of another, and connected together using MIVs, rather than bonding two fabricated dies together using micro-bumps and/or TSVs. Since the pitch of TSV is limited by the micro-bump pitch as well as the alignment accuracy, TSV-based 3D technology, which is ideal for integration discrete bond components onto a single package, does not provide sufficient integration density to solve the on-chip interconnect problem. In contrast, due to the extremely small size of MIVs, monolithic 3D integration scheme enables orders of magnitude of higher integration density than TSV-based 3D technology [60]. There are two approaches to realize monolithic 3D integration:

- Depositing a fresh layer of silicon on top of interlayer dielectric to form a new surface for active devices.
- Transferring a layer of high-quality silicon or even completed devices from a sacrificial wafer onto the primary wafer.

However, the state of this technology is still not mature enough to realize reliable high-performance 3D MIC at present. The reasons can be attributed to: i) its complex processing leading to inferior quality of devices in the upper planes, and ii) limited number of layers due to thermal constraints [61]. In general, monolithic 3D ICs are still a developing technology and are considered by most to be several years away from production.

In this dissertation, we focus on a particular subset of 3D SIC in which the vertical interconnects are realized by TSVs. Therefore, the term "3D IC" in this dissertation refers to such kind of 3D integration scheme unless otherwise specified. As the critical enabler for this type of 3D integration, TSV plays an important role, and its fabrication issues will be discussed next.

2.3 Fabrication of TSVs

TSV is nothing but a hole that goes through the silicon substrate filled with a conducting material (e.g., copper or tungsten). The key design attributes of TSVs (such as the diameter, pitch, aspect ratio [1], conducting material, etc.) are determined by the stage which they are constructed at. In general, TSVs can be manufactured using the following three different processes, including via-first, via-middle, via-last processes, as shown in Figure 2.12.

2.3.1 Via-first process

In this process, TSVs are fabricated before the transistor are patterned in silicon (i.e., prior to *Front-End of Line* (FEOL)) [63, 64]. Key steps in this process are as follows:

[1] Aspect ratio is the ratio of TSV diameter to its depth

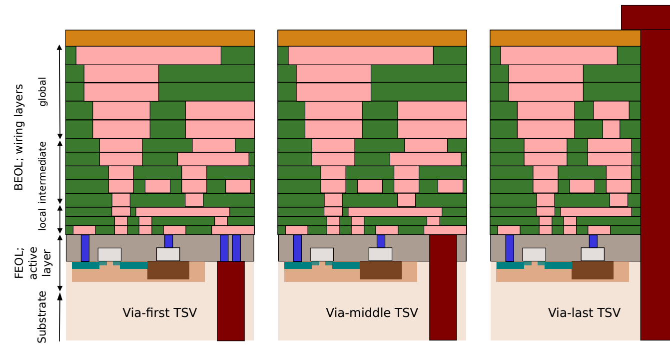


Figure 2.12: Illustration of via-first, via-middle, and via-last TSV technology process [62].

- First, the silicon wafer surface is patterned and TSVs are etched and filled on the transistor side (front or active side) of the wafer.
- After the TSV formation, the wafer is planarized and readied for transistor creation.
- Next, the creation of transistors is followed by the *Back-End of Line* (BEOL) process, which creates the multiple metal/insulator layers on the silicon wafer.
- Finally, the wafer backside is thinned to reveal the vias and creates a RDL.

An advantage of this process is that only known-good wafers with defect-free TSVs can be used for subsequent steps as the vias are formatted before transistor creation. However, a key limitation of this process is that all the TSV materials and processes need to be compatible with CMOS processes and temperature. Due to a lower-bound temperature constraint dictated by the FEOL processing, via-first process excludes the usage of copper TSVs. As a result, polysilicon vias are mostly used in the via-first process which has a relatively high resistance [65]. In [66], a process was discussed in which high density via are created in a via-first process using a silicon as a sacrificial material during the initial via formation step. After the transistor formation processes are completed, the silicon is replaced with tungsten.

2.3.2 Via-middle process

In this process, TSVs are fabricated after front-end device creation, but before the metallization layers are patterned (i.e., prior to BEOL) [67]. Key steps in this process are listed as follows:

- Front-end devices are fabricated including transistors and several lower metal layers.
- Next, TSVs are created from the active device side typically at temperatures in 400 °C-450 °C.

- Afterwards, the TSV is plated with copper or tungsten.
- The silicon wafer is then annealed at around 400 °C to relieve stresses in the TSV and to stabilize the TSV metal structure.
- After annealing, the silicon wafer is planarized with a typical damascene process and readied for the remaining BEOL metallization.
- The wafer is mounted on a carrier after BEOL metallization is completed, and thinned to reveal the TSVs on the inactive (i.e., back-side) of the wafer.
- Via reveal is optionally followed by the creation of a RDL and back-side bump formation processes which is a critical structure for die-stacking.

An advantage of this process is that chip fabrication processing temperatures are lower than that for via-first process, which can reduce the thermal-induced stress in silicon due to the *Coefficient of Thermal Expansion* (CTE) mismatch between the silicon and TSV metal.

2.3.3 Via-last process

In this process, TSV formation occurs after the metallization layers and transistors are fabricated (i.e., after BEOL) [68, 69], either prior or post thinning. Key steps in this process are as follows:

- First, transistors and the entire BEOL process is completed.
- Next, the silicon wafer is temporarily attached to a carrier by using an adhesive (also referred to as a wafer bond/debond adhesive) on the active (i.e., front-side).
- Last, the wafer is thinned and TSVs are created using a similar process to the via-middle process described in Section 2.3.2. This is followed by the manufacturing of the RDL and back-side bumping.

An advantage of this approach is that the thermal stresses are reduced compared to via-first and via-middle process options. However, there are two significant concerns with this process. First, since TSVs are created after the BEOL, they have to land at the right metal layer. To this end, special landing surface (also referred to as catch-up) is included in metal layers for placing the TSVs precisely, which requires careful design and restricts the placement of TSVs. Secondly, as the TSVs are formed after complete wafer processing, yield loss due to TSV formation could lead to loss of valuable silicon.

Table 7.2 shows some of the published data on TSV diameters, depth, electrical and stress characteristic, which are determined by the choices of TSV materials and manufacturing process. This choice depends on final application requirements and infrastructures in the supply-chain. From the perspective of physical design, via-first TSVs are less intrusive as they interfere only with the transistor, the bottom and top metal layers, whereas via-last TSVs interfere with all metal layers [70], as shown in Figure 2.12.

As a result, via-first TSVs have their landing pads on both the bottom metal layer and the top metal layers, while the landing pads of via-last TSVs are located only on the top metal layers. In order to provide the connection between different TSVs, local interconnect and vias are used in between adjacent tiers for via-first TSVs. However, for via-last TSVs, they just need to be stacked on top of each other. In general, via-first TSVs are usually used for signal delivery (i.e., signal TSVs), whereas PDN normally utilize via-last TSVs (i.e., P/G TSVs).

Table 2.1: TSV dimensions reported in recent literature

Reference	TSV diameter (μ m)	Silicon thickness (μ m)	Aspect ratio	TSV material	Process
[66]	2	45	1:22.5	Tungsten	via-first
[71]	5	150	1:30	Poly-silicon	via-first
[72]	3	100	1:33.3	Tungsten	via-middle
[73]	10	50	1:5	Copper	via-last
[74]	6	55	1:9	Copper	via-middle
[75]	10	50	1:5	Copper	via-middle
[76]	5	50	1:10	Copper	via-middle/via-last
[77]	20	50	1:2.5	Copper	via-last
[78]	2	30	1:15	Copper	No specified
[79]	3	50	1:17	Copper	via-middle

2.4 Reliability Issues Related to TSVs

Despite their importance, TSVs have remained a bottleneck for yield boost and reliability improvement of 3D ICs. Typically, reliability complications in TSVs arise either from stress-related or EM-related sources, which will be discussed as follows.

2.4.1 Stresses in TSVs

The stresses in TSVs come from the following two sources: i) growth stresses, that arise as a result of via-filling by electroplating, and ii) thermo-mechanical stresses that arise due to thermal expansion mismatch between copper in the via and the surrounding silicon.

when the electro-deposited copper grows radially inwards on the copper-seed layer from the via side-walls, growth stress would arise, which often results in a seam along the TSV axis. Such a seam might leave a thin void-line along the TSV axis (as shown in Figure 2.13), which can grow during post-deposition annealing, and lead to mechanical and electrical performance loss. Even at room temperature, electroplated copper can undergo significant grain growth by self-annealing. This growth results in volume shrinkage of the copper TSV by elimination of grain boundaries [80]. Consequently, any existing void would be grew during both self-annealing and pre-chemical-mechanical polishing annealing steps. During annealing, hydrostatic stress gradients induced around pre-existing defects in the copper may be relieved by diffusion of lattice vacancies towards the existing void near the axis, causing the void to be increased [81].

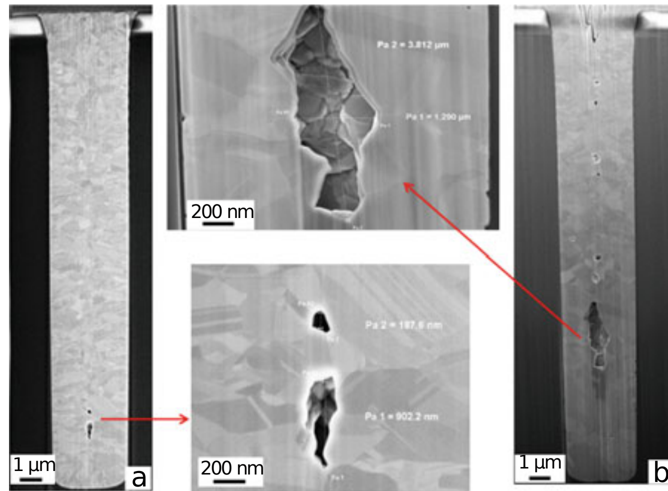


Figure 2.13: SEM images of TSVs showing centerline void after annealing at: (a) 150 °C and (b) 300 °C [81].

Thermo-mechanical stresses arise due to the CTE mismatch between TSV and silicon substrate. Under normal operations, the electronic device experiences heavy thermal cycles, which can leads to accumulation of hydrostatic stress in TSV or filler [82], and TSVs with high impurity levels appear to lead to high residual stresses [83]. In addition, it has been reported in [84] that large tensile radial stress can also exist at the silicon-copper interface [84], which make these sites prone to failure in TSV assemblies. For a given diameter, the radial stress at the interface would generally increase with the length of the TSV [84]. Consequently, the stress may initiate micro-cracks from the interface between a TSV and its dielectric liner, and further propagates them on the silicon substrate surface. Furthermore, the induced stresses near the surface of silicon next to the copper-filled TSVs have an adverse impact on the electrical performance of devices in the immediate neighborhood of the TSVs due to induced piezoresistivity, which results in degradation of carrier mobility [85]. In order to reduce this degradation, a KOZ (i.e., a region in which active devices cannot be placed) is inserted in the immediate vicinity of each TSV. However, the KOZs also introduce significant overheads on the area in the chip for active devices, particularly with the increasing density of TSVs.

2.4.2 Electromigration in TSVs

EM is a significant reliability issue in metallic interconnects in electronics, particularly in BEOL structure. In theory, TSVs are generally less susceptible to EM-induced failures due to their relatively large cross-sections, which reduce the current density. However, with the combination of electric current and complicated stress states above and below the TSV, substantial diffusional effects can be caused in TSVs, which give rise to EM-related void growth. Typically, EM reliability concerns in the interconnects in a 3D packaging can be the micro bumps, TSV, and its connected metal layers including BEOL

and RDL. For TSV and its connected metal layers, EM failures can be inside the TSV close to the connection interface or in the metal layers adjacent to TSV, depending on the interconnect structures and the directions of electron flows.

In [86], two EM testing structures were considered as shown in Figure 2.14, and the EM failure modes in the two testing structures are illustrated in Figure 2.15. For up-stream electron flows, the failures are at the bottom of TSV for the two testing patterns due to the copper flux divergence at the interface of bottom TSV and barrier metal layer. However, because of the reservoir effect for RDL case and the redundancy design of upper die BEOL, no TSV EM damages are observed for down-stream electron flows.

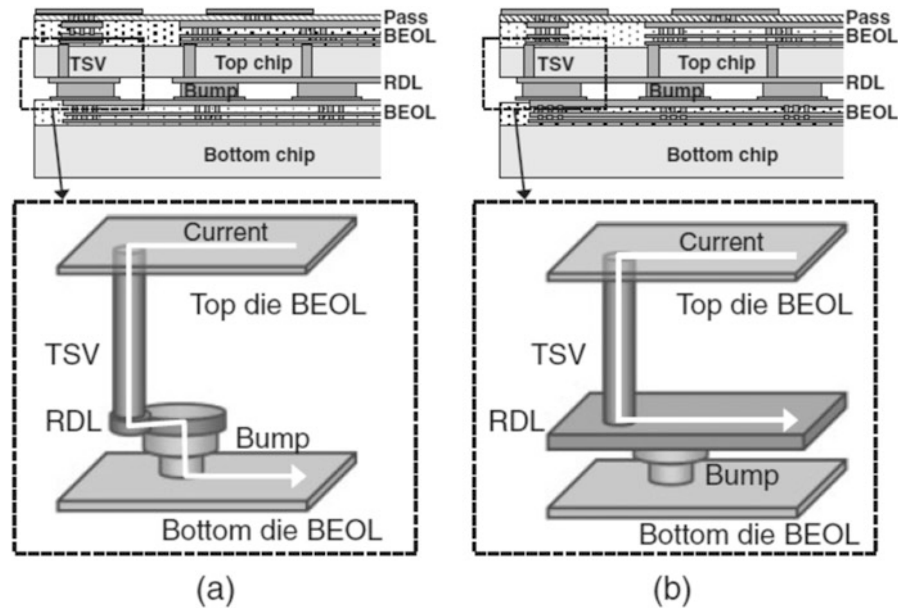


Figure 2.14: Schematic of EM testing pattern: (a) TSV+BEOL and (b) TSV+RDL [86].

EM failure can be also observed within the metal layers connected to TSV, as shown in Figure 2.16. In general, EM experiments on testing structures with thin as well thick metal layers at the top and bottom of TSVs show no void formation due to EM degradation within the TSVs, regardless of the current flow direction [87, 88]. However, the voids can still nucleate at the copper/SiN interface, which is known to be the typical failure interface in copper damascene interconnects [89]. The reason is that the migration of copper atoms out of the void region can occur into the copper lines at the end of the TSVs although the migration from the TSV to the void is prevented by the TiN barrier layer. Therefore, the layer of TiN actually becomes the root cause behind EM void nucleation as a diffusion barrier. In particular, for a thin line, void can occupy the entire thickness of the line, and the voiding effect cannot be mitigated by increasing line width because current crowding at the location where the electron current inside the TSV is greater for the thicker lines. By contract, for a thick line, the voids nucleate at the interface between TiN and the copper line.

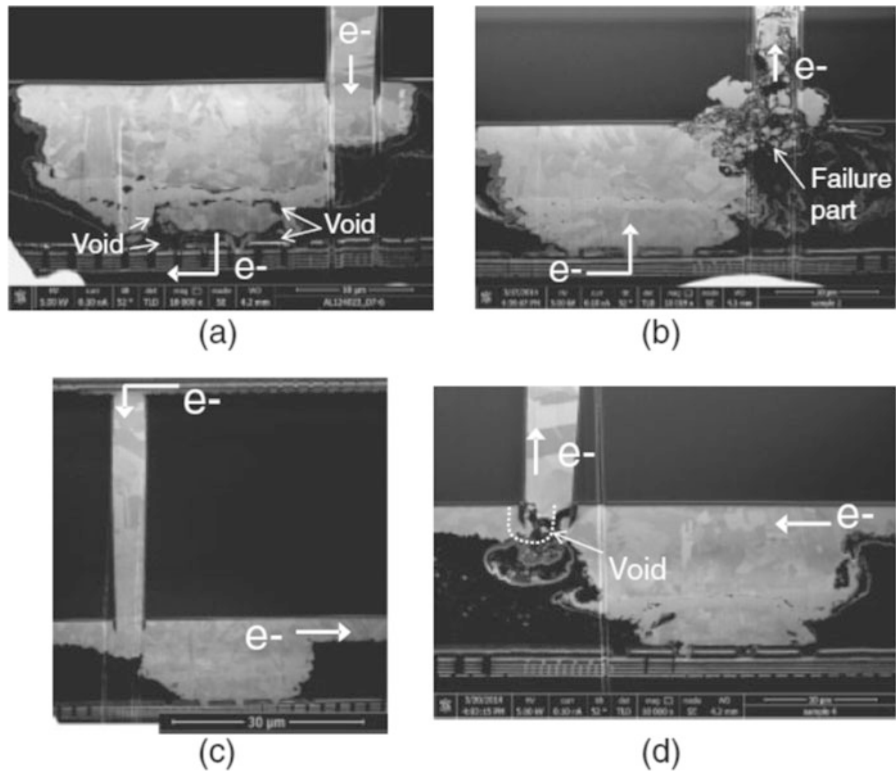


Figure 2.15: TSV EM failure modes: (a) down-stream current with TSV+BEOL, (b) up-stream current with TSV+BEOL, (c) down-stream current with TSV+RDL, and (d) up-stream current with TSV+RDL [86].

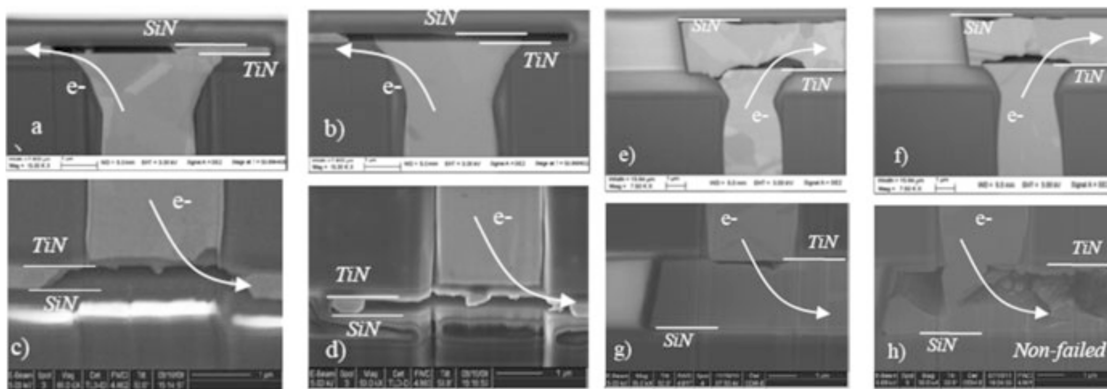


Figure 2.16: EM failure modes in metal line connected to TSV: (a,b) thin line with up-stream current, (c,d) thin line with down-stream current, (e,f) thick line with up-stream current, and (g,h) thick line with down-stream current [87].

2.5 Testing of 3D ICs

Compared to the testing of 2D counterparts, 3D ICs introduce many new challenges for testing. First, during 3D stacking, yield loss for each die in a 3D IC would be compounded, and thus stacking of untested die can lead to prohibitively low product yields. As a result, it is necessary to perform pre-bond testing or the testing of dies prior to being bonded to a 3D stack. This way, only the die that are known to be defect-free can be stacked together. In addition, the dies in the same stack can be chosen based on metrics such as speed or power consumption, which is so-called “die-matching”. Next, it is also important to perform post-bond testing, in which either a partial stack to which all dies have yet to be bonded or the complete 3D stack is tested. Post-bond testing ensures that the 3D stack is functioning as intended and that no errors have been made or new defects have been introduced during manufacturing process. However, both pre-bond and post-bond testing can introduce many DfT challenges, which are not presented in *Two-Dimensional Integrated Circuit* (2D IC) testing.

Pre-bond testing of dies offers many challenges for DfT. First, compared to the non-thinned counterparts, the thinned wafers are far more fragile, and thus a limited number of contacts can be made during probing and low contact-force probes are a necessity. However, due to design partitioning in 3D stacks, a die may contain only partial logic instead of completely functional circuits, which limits the number of tests applied to a circuit. Next, TSVs also introduce challenges for pre-bond testing since their high densities and small sizes make them difficult to probe individually with current technologies. Furthermore, limitations on dedicated test TSVs, over-sized test pads for probing and test signals, and the availability of I/O pins only through one end of the stack can also make design and optimization tool imperative for proper test-resource allocation.

During post-bond testing, a partial stack needs to be tested in order to ensure that no new defects are introduced during stacking. This requires a test architecture and appropriate optimization approaches to reduce test-application time and increase the test availability for partial stack. Next, due to the limited area resource for TSVs, few test TSVs are available to the stack, which also limits the post-bond testing. Furthermore, the limited number of dedicated test pins can only provide test access through one end of the 3D stack. Last, thermal issues should also be considered during post-bond testing. The stacking of substrates and oxide layers greatly increases the difficulty of heat dissipation, particularly for the dies farther from the heat-sink. Therefore, thermal constraints should be taken into account during 3D testing, which restricts the feasibility of test scheduling and motivates low-power pattern generation.

3 Thermal Stress-aware Power/Ground TSV Planning

Among the existing design issues of 3D ICs, reliable power supply design is one of the major challenges. In particular, smaller footprints combined with multiple stacked tiers imply more severe power delivery problems. In addition, due to routing congestion issue, the number of available TSVs for P/G nets is limited. As a result, it becomes a challenging task to deliver enough current to all parts of the 3D stack while guaranteeing high power integrity for the stable operations of 3D ICs, including low voltage drop in *Direct Current* (DC) components (i.e., IR-drop) and low voltage fluctuation in *Alternating Current* (AC) components (i.e., Ldi/dt). In 3D ICs, IR-drop is particularly important since die stacking has a higher impact on IR-drop than Ldi/dt noise [90]. Therefore, in order to enhance power integrity in 3D ICs, 3D PDN has to be carefully designed for IR-drop reduction [91].

On the other hand, the insertion of P/G TSVs can also impact the timing performance of a 3D IC significantly. First, TSV fabrication causes thermal stress around TSVs due to the mismatch in the CTE between silicon and copper[92]. The thermal stress can cause hole and electron mobility variations in the transistors, which can be translated into variation in the gate delay. For an individual gate, TSV stress-induced timing variations can be as much as ±10% [93]. Moreover, because of larger radiuses of P/G TSVs, compared with signal TSVs, they result in more delay variations [94]. Second, the insertion of P/G TSVs also changes the supply voltage of each node in the PDN [95], impacting the delays of connected gates. The rapid transient current flow in the PDN causes spatial and temporal voltage drops, reducing the supply voltage V_{dd}. The reduced V_{dd} decreases the gate drive strength, thereby increasing the gate delay [96].

Recently, several techniques have explored P/G TSV planning [90, 97–99], but most of them focus only on relieving voltage drop in PDN, and the timing optimization is ignored. On the other hand, several methods have studied TSV-induced stress with the focus on the stress-induced timing violations. In [93], a design flow is proposed to analyze timing variations due to TSV-induced stress. In [100], the authors demonstrated that the amount of path delay variations depends on the relative location of the gates to the TSVs, and proposed a set of layout guidelines to optimize delay variations. However, in these work, the TSVs were assumed to be regularly placed in advance. The timing optimization only relies on the relocations of gates on the critical paths. In [101], a force-directed 3D TSV placement algorithm was proposed, by considering the effect of TSV-induced stress on carrier mobility variation, in which TSV and logic gate locations are determined simultaneously. However, only the impact of TSV-induced stress on circuit delay were considered in these previous work. As an important factor which

changes the gate delay, the voltage drop should also be taken into account for timing optimization, especially during the P/G TSV planning.

In the traditional 2D IC design methodology, PDNs are synthesized early in the design flow. Since the exact placements of gates are not known, designers typically resort to prior experience or rough estimation based on incomplete information. The PDN is created during floorplanning, which is so-called “floorplan and PDN co-synthesis” [102]. In [103], a 3D IC floorplan and PDN co-synthesis tool was proposed. During the co-synthesis process, the IR-drop and the routing area of the P/G wires are minimized at the same time. However, the P/G TSVs were regularly distributed for each layer due to the incomplete power profile. Compared to the regular one, the irregular TSV distribution topology provides a superiority in terms of power supply noise, TSV number, and footprint area [95, 104, 105]. Therefore, a novel design methodology was proposed in [95], in which the planning of P/G TSVs was performed on a detailed layout. The “post-layout” P/G TSV planning can insert the TSVs into the optimal locations for IR-drop reduction, but also could result in extra delay variation due to the TSV-induced stress and voltage drop, which is ignored in the previous methodologies.

In this chapter, we propose a stress-aware P/G TSV planning to accurately analyze and then optimize the circuit delay affected by the post-layout insertion of P/G TSVs due to their induced stress and IR-drop. The key contributions in this chapter include:

- We present a systematic and automatic flow for TSV placement and sizing to optimize P/G TSV planning by considering both voltage- and stress-induced delay variations
- We propose a *Mixed Integer Bilinear Programming* (MIBLP) to optimize the PDN synthesis.

The rest of this chapter is organized as follows. First, the compact TSV-induced stress and mobility variation models and 3D PDN design are reviewed in Section 3.1. Section 3.2 gives the motivation of our approach. Section 3.3 describes our MIBLP-based methodology. In Section 3.4, we report our experimental results. Finally, conclusions are drawn in Section 3.5.

3.1 Preliminaries

3.1.1 Power delivery network in 3D ICs

In 3D ICs, the supply power is fed from the package through power bumps and is distributed over the bottom-most layer via on-chip PDN, which is based on a grid structure [103], as shown in Figure 3.1(a) To reach upper layers, the supply power travels through P/G TSVs that connect different layers, as shown in Figure 3.1(b). In the PDN, the pitch determines the distance between P/G line, and the width determines the thickness of each power wire. For each layer in a 3D IC, there is an individual 2D power grid. P/G TSVs are used to connect the 2D grids of each layer.

In general, there are two types of 3D P/G network topologies: uniform and non-uniform grids. In a uniform mesh the pitches in different layers are the same, while in non-uniform topologies different layers can have different layouts. Without loss of generality, we use a uniform mesh to construct the global mesh in this chapter, where the pitch of each tier's 2D mesh is equal. However, our proposed methodology can also handle non-uniform network topologies. Moreover, since we focus on the IR-drop in this chapter, the PDN is optimized using a resistive model, where the branches and TSVs are modeled as resistances.

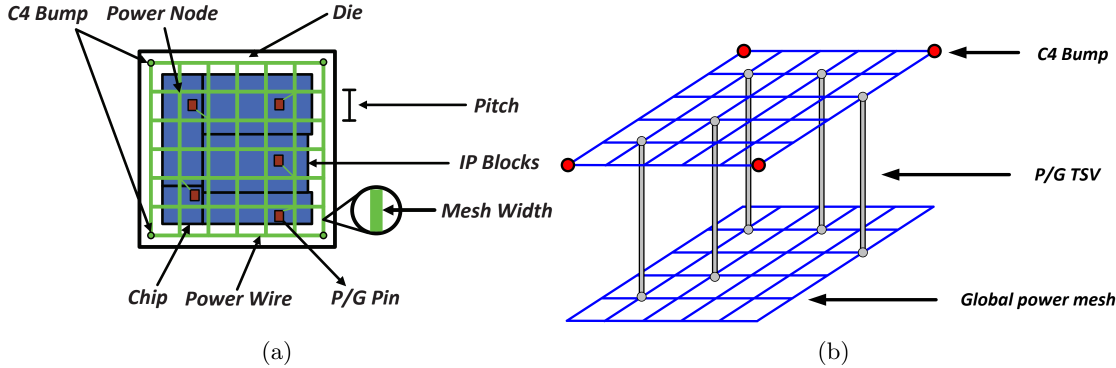


Figure 3.1: (a) On-chip power distribution network. (b) Three-dimensional uniform power distribution network.

Once the PDN is created, it needs to be analyzed by calculating all the voltages at the nodes and estimating the IR-drop. In the IR-drop analysis, the PDN can be modeled as a linear resistive system $\mathbf{G}_{3d}\mathbf{V} = \mathbf{I}$. Here $\mathbf{G}_{3d}$ represents the conductance matrix of the power grid for the interconnected resistors in 3D ICs; $\mathbf{V}$ is a vector containing all the unknown node voltages; and $\mathbf{I}$ is a vector of current loads. Similar to [106], we construct the 3D conductance matrix $\mathbf{G}_{3d}$ using the 2D conductance matrix $\mathbf{G}_{2d}$ and the TSV conductance matrix $\mathbf{G}_{tsv}$. The conductance matrix $\mathbf{G}_{3d}$ of a two-layer 3D IC, each with a power grid containing N rows and N columns, can be expressed as follows:

$$\mathbf{G}_{3d} = \begin{pmatrix} \mathbf{G}_{2d_1} + \mathbf{G}_{tsv_{1,2}} & -\mathbf{G}_{tsv_{1,2}} \\ -\mathbf{G}_{tsv_{1,2}} & \mathbf{G}_{2d_2} + \mathbf{G}_{tsv_{1,2}} \end{pmatrix} \in \mathbb{R}^{2N^2 \times 2N^2} \tag{3.1}$$

where $\mathbf{G}_{2d_i} \in \mathbb{R}^{N^2 \times N^2}$ $(i = 1, 2)$ is the conductance matrix on layer i without TSVs. $\mathbf{G}_{tsv} \in \mathbb{R}^{N^2 \times N^2}$ represents the conductances due to the existence of TSVs and can be written as:

$$\mathbf{G}_{tsv} = diag(g_{tsv_1}, g_{tsv_2}, ..., g_{tsv_m}, ..., g_{tsv_{N^2}}) \tag{3.2}$$

where g_{tsv_m} is the conductance of the TSV placed at the mth TSV candidate, which is connected between node m and node $m + N^2$ in the 3D PDN [1]. It is similar for multiple

[1] In this chapter, we use the following notation methods to describe a node in a 3D PDN: if a

layers 3D IC by apply Equation (4.17) pair-wisely between adjacent layers [104]. By solving this linear system $\mathbf{G}_{3d}\mathbf{V} = \mathbf{I}$, we can obtain the unknown voltage of each node in the PDN with the current vector $\mathbf{I}$.

Once the PDN without P/G TSVs is created on each layer, the matrix G_{2D} is constant. Hence, the conductance matrix of 3D ICs G_{3D} depends only on the matrix of TSVs G_{TSV}, which itself is highly dependent on the location, size and the number of P/G TSVs. First, different TSV locations and numbers will affect the position and number of non-zero elements on the diagonal of G_{TSV}. Second, the non-zero values in G_{TSV} depend on the TSV sizes (in this chapter, the TSV diameter). Based on the analytical expression of the TSV resistance given in [107], the conductance of a TSV is:

$$g_{tsv_{\mathrm{DC}}} = \frac{\pi r_{tsv}^2}{\rho h_{tsv}}, \tag{3.3}$$

where ρ is the resistance of the conducting material. h_{tsv} and r_{tsv} represent the height and radius of the TSV, respectively. Conventionally, the TSV height is always equal to the die thickness [108]. Hence, the conductance of one TSV only depends on the size of the TSV for a specific TSV fill material. As a result, different TSV sizes will also impact the 3D PDN. Thus, it is necessary to consider the location, size and number of P/G TSVs simultaneously during the 3D PDN design.

3.1.2 TSV-induced mobility variation

To take TSV-induced stress into account in the timing optimization, an analytical model of TSV stress-induced carrier mobility variation was proposed in [93]. The mobility change due to thermal stress depends not only on the intensity of the applied stress, but also the orientation between the stress and a transistor channel. If a transistor is located in the overlap zone of multiple TSVs, each TSV works as a stress source. The effect from multiple TSVs can be combined by using linear superposition. Total mobility variation of transistor j $(\Delta\mu_j/\mu_j)$ is defined in [93] as

$$\frac{\Delta\mu_j}{\mu_j} = \frac{\Pi B \Delta\alpha \Delta T}{2} \sum_{i \in \mathrm{TSV}(j)} \left(\frac{R_i}{r_{ij}}\right)^2 \alpha\left(\theta_{ij}\right) \tag{3.4}$$

Here Π is a coefficient indicating the sensitivity of mobility change to applied stress, B is biaxial modulus, $\Delta\alpha$ is the CTE difference between copper and silicon, ΔT is the temperature difference between copper annealing and operating temperature. TSV (j) represents the set of TSVs that influence transistor j, R_i is the radius of TSV i, r_{ij} is a distance from the center of TSV i to the transistor j, and θ_{ij} is defined as the angle between the center of the TSV i and the center of the transistor channel of transistor j when it is placed vertically, as shown in Figure 3.2. Note that, for each TSV, there is a KOZ surrounding it. KOZ is the region around the TSV from which all gates must

node at the ith row and the jth column in the kth layer power grid, then we notate it as node $\left(i \cdot N + j + (k-1) \cdot N^2\right)$ in the 3D PDN

"keep out" so that they are not influenced by the TSV-induced stress to some extent. In realistic designs, the KOZ is defined to ensure that the mobility variation in any transistor around an isolated TSV is below preset tolerance $c\%$ [101].

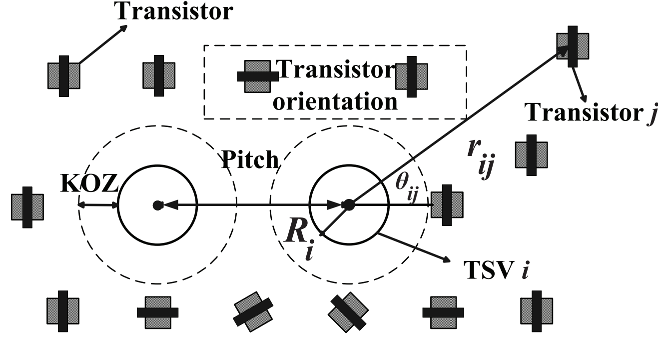


Figure 3.2: Illustration of TSV layout parameters.

For the orientation factor α in Equation (3.4), its value depends on θ and also the type of transistor. The value of θ can be obtained by Equation (3.5):

$$\theta = \tan^{-1}\left|\frac{X_{TSV} - X_{poly}}{Y_{TSV} - Y_{poly}}\right| \tag{3.5}$$

where (X_{TSV}, Y_{TSV}) and (X_{poly}, Y_{poly}) indicate the coordinates of TSV center and transistor channel. Figure 3.3 gives the values of α for different types of transistors and θ values. As shown in it, we can see that hole mobility increases in a vertical region (mobility enhanced zone), while it decreases in a horizontal direction (mobility degraded zone). 45° has no hole mobility change. For electron mobility variation, horizontal direction has more mobility enhanced zone.

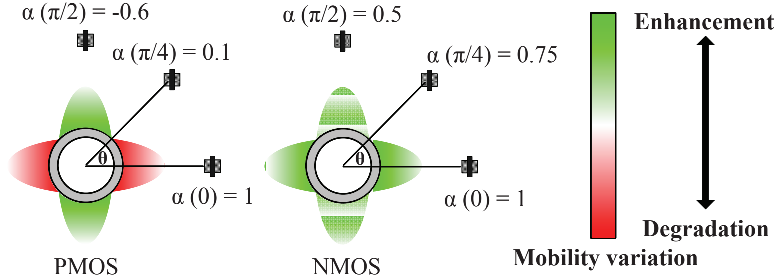


Figure 3.3: Carrier mobility variation surrounding TSVs.

3.2 Motivation

Due to the strong layout dependency, it is possible to take advantage of TSV-induced stress for timing optimization by TSV-stress aware layout optimization, such as gate relocation and TSV placement. In [93, 100], some layout guidelines have been proposed to optimize the stress-induced delay variation with regular TSV distribution topologies.

The basic idea is to relocate the gates on the critical and near critical paths in mobility enhanced zone. Figure 7.2(a) shows how gate relocation works for timing optimization. Gate 1 is electron critical gate because the timing arc is falling on the path [93]. Therefore, the gate is relocated to be placed close to blue area, which is the electron enhanced zone. However, gate 2 is pulled to hole mobility enhanced zone (i.e., the green area in Figure 7.2(a)) because it is hole critical gate. In [101], an irregular stress-aware TSV placement algorithm has been proposed, in which TSV and logic cell locations are determined simultaneously, as shown in Figure 7.2(b). It has been demonstrated that design with irregular TSV topology has superior performance in terms of wirelength and mobility variation, compared to the regular one [101]. However, the impact of IR-drop on gate delay was ignored in the previous work.

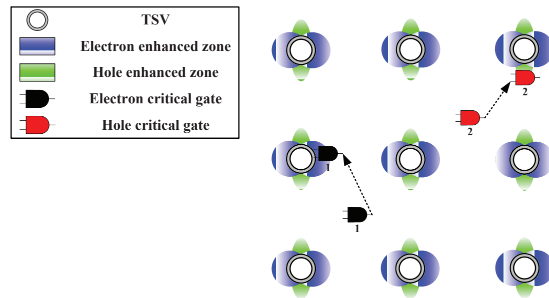
Besides the TSV-induced stress, the IR-drop can also impact the gate delay: a reduced supply voltage deteriorates the gate drive strength, thereby increases the gate delay [96]. As shown in Figure 3.4(c), gate 1 is relocated from grid A to grid B following the stress-aware layout optimization, while gate 2 is moved from grid C to grid D. The relocation of gate might change its connected node of the power grid, and hence the supply voltage which drives it. Without considering the negative impact of IR-drop, there is no guarantee that we can benefit from the gate relocation in terms of gate delay. The previous work demonstrate that an optimized P/G TSV planning can relieve IR-drop in 3D PDN [90, 97–99], which could also be exploited for timing optimization. In this chapter, we consider the TSV-induced stress and IR-drop simultaneously during the P/G TSVs planning. With the ILP method, we can optimize the circuit delay impacted by P/G TSV insertion.

3.3 Methodology

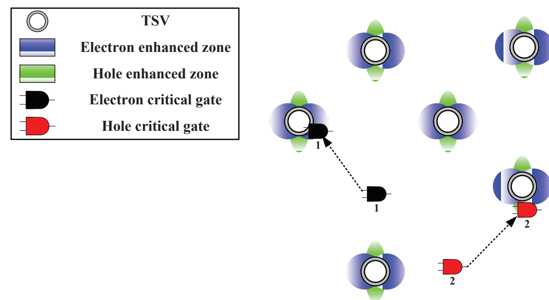
3.3.1 Problem statement

Our objective is to optimize (minimize) the circuit delay impacted by the insertion of P/G TSVs, subject to: the IR-drop ΔV_{dd} of each node in the PDN be within an acceptable range. Due to the TSV-induced stress and IR-drop, the insertion of P/G TSVs could generate delay change. The output of this ILP optimization is a refined P/G TSV planning (including the location and radius of each inserted TSV), which results in the minimal circuit delay taking the P/G TSV-induced delay change into account.

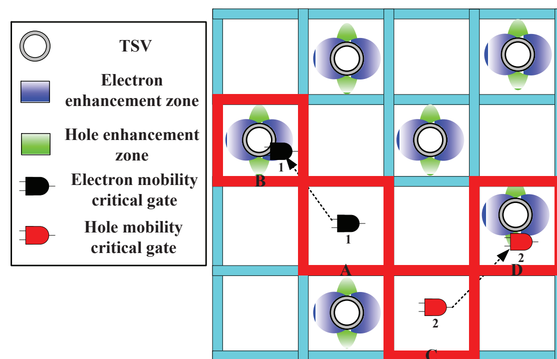
To accurately take this into consideration, delays of all critical and near-critical paths (i.e., *Vulnerable Paths* (VPs)) have to be considered. It should be noted that the P/G TSV-induced delay change of a near-critical path p_i $(\Delta D(p_i))$ could be more than that of a critical path p_j $(\Delta D(p_j))$ such that $D(p_i) + \Delta D(p_i) > D(p_j) + \Delta D(p_j)$. However, in the realistic design, there could be lots of VPs since the paths are fully balanced. Therefore, it is infeasible to take all the VPs into account in the ILP formulation. A solution to this issue lies in the selection of only a small set of *Representative Critical Paths* (RCPs) from the numerous VPs [109, 110]. Based on the method proposed in [110], We start the ILP formulation with the RCPs, which are selected from VPs. Thus,



(a) Gate relocations with regular TSV topology



(b) Gate relocations with irregular TSV topology



(c) Gate reolcations with voltage drop consideration

Figure 3.4: Gate relocations with regular and irregular TSV topologies

the ILP formulation can be expressed by:

$$\begin{aligned}&\text{Minimize}: \max_{j=0}^{N} \sum_{\forall g_{ji}\in \text{RCP}_j} \left(D_0\left(g_{ji}\right) + \Delta D\left(g_{ji}\right)\right)\\&\text{Subject to}: \Delta V_{dd}^{n} \leq \Delta V_{dd}^{\max}, \forall n\end{aligned} \tag{3.6}$$

where j is an RCP, N is the number of RCPs, and g_{ji} is gate i in RCP_j. $D_0\left(g_{ji}\right)$ is the original delay of gate g_{ji}, and $\Delta D\left(g_{ji}\right)$ is P/G TSV-induced delay change. ΔV_{dd}^{n} is the IR-drop of node n in the power grid, and $\Delta V_{dd}^{\max}$ is the acceptable limit of this drop.

3.3.2 Power/ground TSV-induced delay change

In general, the P/G TSV-induced delay change, ΔD, can be divided into two components: i) mobility variation-induced delay change ΔD_{μ}, and ii) IR-drop-induced delay change $\Delta D_{V_{dd}}$. In this work, ΔD can be represented as a sensitivity model:

$$\Delta D = \left(\frac{\partial D}{\partial \mu}\right) \Delta \mu_{TSV} + \left(\frac{\partial D}{\partial V_{dd}}\right) \Delta V_{dd} \tag{3.7}$$

where $\partial D/\partial \mu \left(\partial D/\partial V_{dd}\right)$ is the sensitivity of the delay to mobility (supply voltage) changes. Due to the nonlinearity of the sensitivity function, it is stored as a *Look-up Table* (LUT) rather than a constant sensitivity value. Here, $\Delta V_{dd} = V_{dd} - V_i$. V_{dd} is the nominal supply voltage, and V_i is the voltage of the node connected by this gate. $\Delta \mu_{TSV}$ is the mobility change due to TSV-induced stress, which is obtained using Equation (3.4).

In a realistic 3D IC design, TSV radius is an important parameter in this cylindrical via for the designer or manufacturer when considering the TSV density [108]. Moreover, the various radius can also affect the TSV-induced stress [94]. Therefore, we extend Equation (3.4) to express the P/G TSV-induced mobility variation for each gate j on the RCPs:

$$\begin{aligned}&\frac{\Delta \mu_j}{\mu_j} = \frac{\Pi B \Delta\alpha \Delta T}{2} \sum_{i\in \text{TSV}(j)} \frac{\alpha\left(\theta_{ij}\right)}{r_{ij}} R_{ik}^2 x_{ik}\\&\sum_{k=1}^{n} x_{ik} \leq 1, \quad x_{ik} \epsilon \left\{0,1\right\}\end{aligned} \tag{3.8}$$

where $\text{TSV}\left(j\right)$ is the set of TSV candidates that could influence gate j, working as stress sources. In this work, we assume that there are n different options of TSV radiuses available for the design, and R_{ik} is the radius of the TSV with the kth option. Note that x_{ik} is a binary variable showing if there is a TSV placed in the ith candidate with k^{th} radius option. In this way, we can extract the mobility variation expression for each gate. Based on Equation (3.6), (3.7) and (3.8), we can obtain the objective function by summing all the gate delays on each RCP.

3.3.3 Voltage constraints

For each node n in the power grid, the IR-drop ΔV_{dd}^{n} should be within acceptable limit $\Delta V_{dd}^{\max}$, which is given as

$$\Delta V_{dd}^{n} = V_{dd} - V_i \leq \Delta V_{dd}^{\max} \tag{3.9}$$

Here, the voltage variable V_i represents the voltage of each node in the power grid, which can be obtained by solving the linear system $\mathbf{G}_{3d}\mathbf{V} = \mathbf{I}$.

For the matrix $\mathbf{G}_{3d}$, we can construct it as shown in Section 3.1.1. Because of the various radius options of TSV, the conductance of TSV g_{tsv} in Equation (3.2) is also variable [107]. Similar to Equation (3.8), the conductance of TSV placed at the ith TSV candidate can be written as:

$$\begin{aligned} g_{tsv_i} &= \sum_{k=1}^{n} \frac{\pi R_{ik}^2}{\rho h_{tsv}} x_{ik} \\ \sum_{k=1}^{n} x_{ik} &\leq 1, \quad x_{ik} \epsilon \{0,1\} \end{aligned} \tag{3.10}$$

Here ρ is the resistance of the conducting material, and h_{tsv} represents the height of the TSV. Note that we use the same binary variable x to express the various conductance as in Equation (3.8). Due to the existence of binary variables in $\mathbf{G}_{3d}$, it is very time-consuming to solve this linear system directly. Thus, we use the equations in the system directly as voltage constrains for our ILP model.

For the uth equation in this system, we can get the following constraint:

$$\sum_{v=1}^{LN^2} G_{3d_{uv}} V_v = I_u \tag{3.11}$$

where L is the number of layers in 3D IC, and N is the number of rows (columns) of power grid for each layer. Here $G_{3d_{uv}}$ is the element at the uth row and the vth column in G_{3d}, V_v is the vth element in $\mathbf{V}$ and I_u is the uth element in $\mathbf{I}$.

Due to the special structure of G_{3d} and the existence of binary variables, there are some nonlinear terms in Equation (3.11), such as $g_{tsv_v} V_v$, which can be expanded as

$$g_{tsv_v} V_v = g_1 x_{v1} V_v + g_2 x_{v2} V_v + ... + g_k x_{vk} V_v \tag{3.12}$$

where g_i $(i = 1, 2, ..., k)$ is conductances of TSVs for different radius. In Equation (3.12), both x and V are variables. Due to the existence of bilinear terms in the problem, the ILP problem is an MIBLP problem. Thus, we need to linearize the bilinear terms first. In this work, we replace them by McCormick envelope [111] to linearize these bilinear terms.

3.3.4 Power and stress profiles redistribution

Since we consider the post-layout P/G TSV planning in this work, the proposed approach is performed based on a detailed layout, which means that signal TSVs and the logic gates are already placed. Because signal TSVs have been optimized at their locations to improve wirelength and timing objectives, we avoid overlapping between them. If an optimal P/G TSV location overlaps with gates, we first apply a line-sweeping-based algorithm to find the whitespace [112] in the same layer, and move these gates to the nearest whitespace, which do not overlap with other gates and P/G TSV candidates.

Due to the gate relocations, the power and TSV-induced stress profiles will also be redistributed, which should be taken into account in the ILP formulation. For the power consumption, as shown in Figure 3.5, due to the insertion of TSV t_i at node i, the overlapped gate a have to be moved to the nearest whitespace, which could change the power consumptions of related nodes in the power grid. In this case, because of its relocation, gate a will connect to node j, which is the closest node to the gate after the relocation. Thus, there will be a current, which represents the power consumption of gate a, flowing out from node i and flowing into node j.

For the ith TSV candidate, if we insert a P/G TSV with the kth radius option at this location, all the overlapping gates connected to node i and node $i + N^2$ should be relocated, and some of these gates will be connected to another node j in the power grid. We identify these gates, and calculate the “flow-out” and “flow-in” currents in the corresponding nodes, reflected by $-\Delta I_{i,k}$, $-\Delta I_{i+N^2,k}$, and $+\Delta I_{i\to j,k}$. Then we have

$$\begin{aligned} I_i &= I_i^0 - \Delta I_{i,k} \cdot x_{ik} \\ I_{i+N^2} &= I_{i+N^2}^0 - \Delta I_{i+N^2,k} \cdot x_{ik} \\ I_j &= I_j^0 + \Delta I_{i\to j,k} \cdot x_{ik} \end{aligned} \tag{3.13}$$

Here I_i^0, $I_{i+N^2}^0$ and I_j^0 are the original current sources connected to the nodes i, $i + N^2$ and j, respectively. $-\Delta I_{i,k}$ and $-\Delta I_{i+N^2,k}$ are the currents flowing out from the nodes i and $i + N^2$, while $+\Delta I_{i\to j,k}$ is the current flowing from node i into node j due to the relocation. In this way, we can handle the issue of power profile redistribution in our ILP formulation.

Moreover, if the relocated gate is on an RCP, we will also recalculate the P/G TSV-induced delay change by Equation (3.7) based on its new location, and update the corresponding term in the objective function. Note that, the nonlinear terms in the objective function, which are generated by the recalculation, can also be linearize by McCormick envelope [111].

3.3.5 Scalability

Scalability has always been a critical issue for ILP-based methods [113]. The size of the formulation, which includes the number of variables and constraints, reflects the complexity of ILP formulation. As explained below, our approach offers a very good scaling behavior for circuits with increasing gate count.

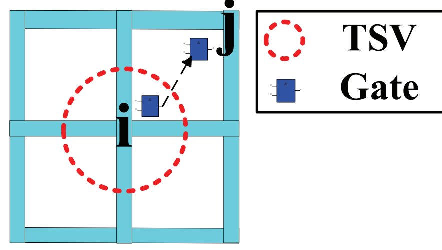


Figure 3.5: The redistribution of gates due to the insertion of TSV.

Suppose we consider a 2D circuit with the number of gates N_{gate}, and partition it into an L-layer 3D IC. Thus, the number of gates for each layer is roughly N_{gate}/L. Assuming that each node in the power grid can drive s gates at most, the number of P/G TSV candidates, N_{TSV}, is equal to

$$N_{TSV} = \frac{(L-1)}{L}\frac{N_{gate}}{s} \tag{3.14}$$

If there are k options for the TSV radius, the number of the variables N_{var} in the ILP formulation is equal to

$$N_{var} = \left(1 + k\frac{L-1}{L}\right)\frac{N_{gate}}{s} \tag{3.15}$$

In addition, the number of total constraint N_{constr} is equal to

$$N_{constr} = N_{RCP} + \left(1 + \frac{L-1}{L}\right)\frac{N_{gate}}{s} \tag{3.16}$$

where N_{RCP} is the number of RCPs. Thus, both the number of variables N_{var} and the number of constraint N_{constr} scale linearly with the number of gates. Thus, we can expect a good scalability of our formulation for larger circuits.

3.4 Experimental Results

3.4.1 Experimental setup and flow

Figure 5.4 shows the implementation flow of the proposed MIBLP-based methodology. In our experiment, we assume a three-layer 3D IC. The benchmark circuits are selected from ITC'99 benchmark suite [114] and the experiments are conducted using the Nangate 45 nm library [115]. The experiments were performed on a server with four AMD Opteron 6174 and 256GB RAM. In this work, we assume that a via-last approach is used in 3D IC fabrication, and we consider three different P/G TSV radiuses: $2.5\mu m$, $5\mu m$, and $10\mu m$ [116]. For the temperature, we assume that ΔT in Equation (3.8) is $175°C$ in this work, which is the case of $25°C$ for the room temperature and $200°C$ for the copper annealing temperature [117].

As shown in Figure 5.4, at first, the netlist and the layout of the 2D ICs are extracted using Synopsys Design Compiler and Cadence SoC Encounter, respectively. UCLA 3D

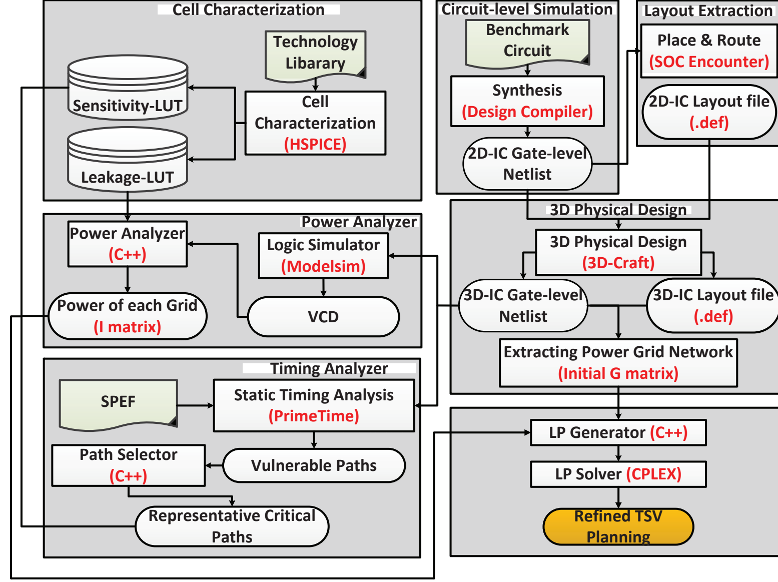


Figure 3.6: The detailed implementation flow of the proposed methodology

physical design flow "3D-Craft" is exploited to convert the 2D IC into a 3D IC [118]. Based on the netlist of each layer, we make a top level netlist that instantiates the layers and connects them using wires which correspond to signal TSV connections. Then we prepare a top level parasitic extraction file (*Standard Parasitic Exchange Format* (SPEF)) for the TSV connections. With a proper timing constraints file, we can run *Static Timing Analysis* (STA) (PrimeTime) and get the list of VPs for the 3D IC. Assisted by the path selector, we select the RCPs from the VPs list.

To obtain the current vector $\mathbf{I}$, we need to calculate the current drawn from each node, by adding the dynamic and leakage currents of all the gates connected to it. For the leakage current, all standard cells in the library are characterized by accurate SPICE simulations to obtain the corresponding leakage current. This information is then stored in leakage LUT. For the dynamic current, we utilize a logic simulator, the circuit netlist, and a set of representative input vector to obtain signal probabilities and activity factors for all the internal nodes in the circuit. Then we send these information into a power analyzer to obtain the power consumptions of each node, and then the current vector $\mathbf{I}$.

In order to generate the objective function in ILP formulation, we first get the exact location of each gate on each RCP from the 3D layout, and calculate the mobility variation by Equation (3.4). By characterizing the standard cell library, we get the mobility and supply voltage sensitivities, and store them in the sensitivity-LUT. Then based on the sensitivity-LUT and RCPs list, we obtain the objective function by Equation (3.7).

Then, we generate the constraints in the ILP formulation. For the linear system $\mathbf{G}_{3d}\mathbf{V} = \mathbf{I}$, the 3D netlist and the 3D layout are used to constructed the initial conductance matrix $\mathbf{G}_{3d}$. After calculating the current vector $\mathbf{I}$ by power analyzer, we can obtain the constraints by Equation (3.11). Finally, the input file for the ILP solver is generated and CPLEX [119] is used to obtain the planning of P/G TSVs.

3.4.2 Impact of design parameters on delay variation

The KOZ of TSV and the PDN pitch are important 3D IC PDN parameters, which are worth considering them during the PDN design. To illustrate the impact of KOZ and PDN pitch on the circuit delay, we investigated benchmark b17 with four different KOZs and pitches, respectively. As shown in Figure 3.7, the circuit delay variation $\Delta D/D_0 = (D - D_0)/D_0$, increases monotonically with the PDN pitch. For a refined power grid, although ΔD_μ might increase due to the increasing number of inserted TSVs, $\Delta D_{V_{dd}}$ could benefit from the refined PDN. However, for the KOZ, delay variation exhibits a "U" shape, and there is an optimal point (in this example at KOZ: 30%). The reason is that although the stress-induced delay change will decrease with the bigger KOZ, the number of TSV candidates will also decrease due to the increased probability overlapping with the fixed signal TSVs, which deteriorates the IR-drop.

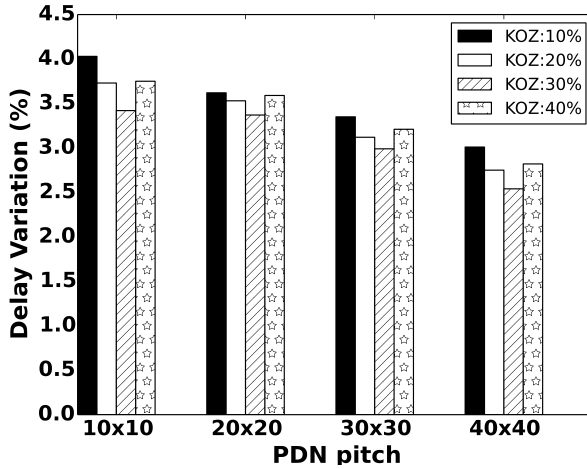


Figure 3.7: Impact of KOZ and PDN pitch on induced delay variation

3.4.3 Comparison with previous technique

To illustrate the significant impact of IR-drop on the circuit delay, we compare our proposed approach with the previous technique, in which only stress-induced delay variation was considered. We investigated a three-layer 30×30 power grid on four different benchmark circuits. Without loss of generality, we set the tolerance value of KOZ $c = 25\%$.

Table 3.1 compares the results obtained from our proposed technique to the method presented in [93]. In the previous work [93], the authors claimed that it is possible to take advantage of TSV-induced stress for timing optimization due to its strong layout-dependence. By TSV-stress aware layout optimization, the critical path delay can be reduced. However, because of ignoring the impact of IR-drop on circuit delay, the previous method is too optimistic to overestimate the effects of these methods on the circuit delay optimization. As shown in Table 3.1, the negative values in the fourth column mean that the critical path delay can be reduced by the TSV-stress aware layout optimization without considering the voltage variation. However, the IR-drop, which

always has a negative impact on circuit delay, could be more critical compared to the impact of stress variation on the circuit delay. For example, for b17, by taking the voltage variation into account, the circuit delay could be increased to 4.328ns, while the circuit delay would be reduced to 4.117ns without the consideration. The IR-drop can lead to significant (close to 8% in our test case) underestimation of circuit delay.

Table 3.1: Comparison with previous technique (only considering TSV-induced stress)

Benchmark Circuit	Original Delay(ns)	Only considering stress [93]		Proposed technique	
		$D\,(ns)$	$\Delta D/D_0$	$D\,(ns)$	$\Delta D/D_0$
b17	4.199	4.117	-1.95%	4.328	3.07%
b18	11.336	11.305	-0.27%	11.946	5.38%
b19	10.057	10.014	-0.43%	10.832	7.71%
b22	8.265	8.281	0.19%	8.719	5.49%

3.4.4 Accuracy and Runtime

Table 3.2 shows the results of the proposed MIBLP-based method. As shown in this table, when we consider P/G TSV-induced delay changes, the delay variations for these benchmark circuits show difference varying from 2.54% to 12.43% for different power grids and KOZs. The proposed technique can obtain the minimum circuit delay within a few minutes. Moreover, as shown in Table 3.2, all the delay variations are positive for different PDN pitches and KOZs. These results further prove that the previous work overestimates the benefit for timing optimization produced by gate relocation. Although the TSV-induced stress could improve the circuit delay by gate relocation, IR-drop can cause more degradation on it. Due to the negative impact of IR-drop, the optimized planning of P/G TSVs can only minimize the detrimental effect on circuit delay.

Table 3.2: Comparison of the original delays with optimized circuit delays considering P/G TSV-induced delay changes

Benchmark Circuit	# of Gates	Original Delay(ns)	PDN	P/G TSV-induced Delay Variation				Runtime (s)
				KOZ : 10%	KOZ : 20%	KOZ : 30%	KOZ : 40%	
				$\Delta D/D_0$	$\Delta D/D_0$	$\Delta D/D_0$	$\Delta D/D_0$	
b17	37117	4.199	10×10	4.03%	3.73%	3.42%	3.75%	2.34
			20×20	3.62%	3.53%	3.37%	3.59%	15.23
			30×30	3.35%	3.12%	2.99%	3.21%	50.27
			40×40	3.01%	2.75%	2.54%	2.82%	179.44
b18	92048	11.336	10×10	7.95%	7.57%	7.26%	7.51%	2.56
			20×20	7.03%	6.34%	6.11%	6.26%	16.77
			30×30	5.89%	5.45%	5.02%	5.57%	89.53
			40×40	5.31%	5.02%	4.52%	5.15%	143.41
b19	174157	10.057	10×10	12.43%	11.49%	10.78%	11.34%	2.77
			20×20	11.69%	10.53%	10.04%	10.41%	20.78
			30×30	8.57%	7.93%	7.21%	7.47%	100.67
			40×40	6.28%	5.75%	5.34%	6.03%	397.46
b22	28317	8.265	10×10	8.04%	7.99%	7.32%	7.56%	2.76
			20×20	7.21%	6.81%	6.27%	6.73%	15.02
			30×30	6.04%	5.71%	5.03%	5.35%	40.78
			40×40	5.78%	4.05%	3.51%	3.72%	130.61

3.5 Conclusion of Chapter

Three-dimensional chip stacking with TSVs has gained tremendous interests for the continuation of the Moore's law. The fabrication of TSVs causes thermal stress, which can lead to significant timing variations. Several technologies have been proposed to take advantage of the TSV-induced stress for timing optimization. However, the negative impact of IR-drop on gate delay was ignored in the previous work. Thus, it is necessary to take the TSV-induced stress and IR-drop into account simultaneously for timing optimization during the P/G TSV planning. In this chapter, an MIBLP-based methodology is proposed to solve this optimization problem. According to our simulation results, the combined effects between TSV-induced stress and IR-drop always degrade the circuit delay. By a refined P/G TSV planning, we can minimize the detrimental effect on circuit delay.

This page would be intentionally left blank.

4 Electromigration-aware Local-via Allocation in Power/Ground TSV

With increasing temperature and current density, EM becomes a major interconnect reliability concern for 3D ICs. Compared to the interconnects in 2D ICs, EM reliability in TSVs become further exacerbated by increasing current density, higher temperature, and thermal mechanical stress [120]. Moreover, as P/G TSVs experience large unidirectional currents, they are more susceptible to EM effect than signal TSVs carrying bidirectional currents. Consequently, the gradual transport of metal atoms caused by EM leads to void nucleation and growth in P/G TSVs, which significantly increase their resistances. This affects the power distribution to the underlying logic circuitry, and may cause harmful voltage drops [121].

A typical PDN of a three-die 3D IC with via-first TSVs is illustrated in Figure 4.1 [1]. Due to the increased power density in 3D ICs, each 2D PDN is typically designed hierarchically, in which global and local PDNs are routed in the top- and low-level metal layers, respectively. As shown, the 3D global PDN consists of the individual 2D global power meshes of all the dies in the 3D IC, and P/G TSVs are used to connect them together for D2D vertical power delivery. However, unlike via-last TSVs which go through all the metal layers, via-first TSVs only have two landing pads on the top-most metal layer (i.e., M_{top}) and the bottom-most metal layer (i.e., M_{bottom}) of adjacent dies. Therefore, it is necessary to place local vias on the M_{bottom} landing pad of each P/G TSV to reach the 2D P/G mesh on the upper metal layers [2]. In other words, P/G TSVs need to use local vias to provide vertical power delivery in 3D PDNs, and both of them have been regarded as EM-prone structures [121–125].

While the EM issue of P/G TSV has been actively studied in the literature [121, 124–126], most of the existing work ignored the existence of local vias inside it, except [126]. In [126], an EM model was proposed by taking the interplay between local vias and P/G TSVs into account. However, this solution ignored the routing cost associated with local vias. Typically, the size of a TSV is gigantic (a few μm) in comparison to a local via (a few tens of nm) [126], and thus it is practical to insert multiple local vias in each P/G TSV for load balancing of the current density per via. By reducing the current density of each local via, EM reliability of P/G TSVs can be improved effectively. However, since local vias need to penetrate several metal layers to reach 2D P/G mesh (as shown in Figure 4.1), they can be blockages in routing stage, in which the routing resources

[1] Since only one type of TSV is preferable in a manufacturing process, here we assume that both P/G TSVs and signal TSVs are via-first type, as in [95].

[2] With via-last TSVs, local vias are unnecessary because TSV landing pads adjoin 2D P/G mesh on the top-most metal. In this chapter, we will limit our scope to the 3D ICs with via-first TSVs.

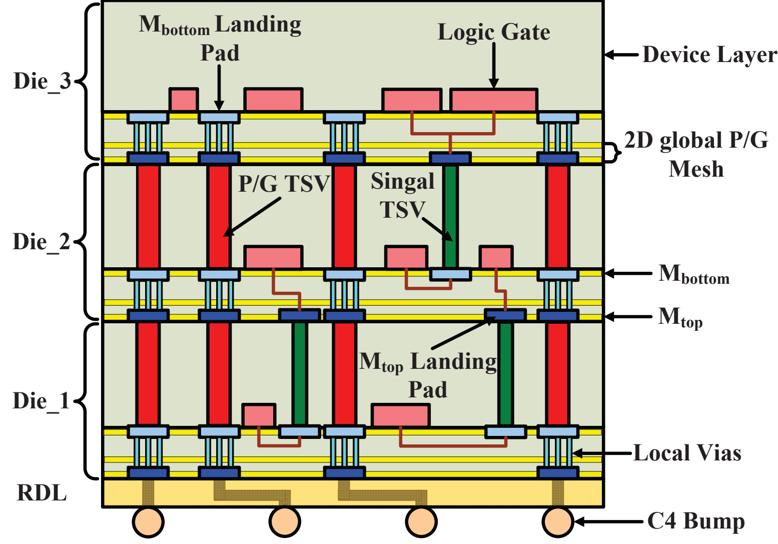


Figure 4.1: PDN of 3D ICs with via-first TSVs.

are either reduced or unavailable. Consequently, excessive local vias may consume too much routing area, which leads to exacerbated congestion problem and increased delay overhead. Therefore, it is imperative to handle the trade-off between EM reliability in P/G TSVs and timing performance of 3D ICs by determining the optimal number of local vias for each P/G TSV.

In this chapter, an efficient post-placement local-via allocation technique is proposed to solve the aforementioned problem. The objective is to minimize the local via-induced routing congestion while satisfying the given requirement of EM reliability. Overall, our contributions are summarized as follows.

- We propose an ILP-based methodology for local-via allocation for P/G TSV in 3D PDNs.
- We investigate the impact of temperature, 3D design partitioning, and target EM-related TSV lifetime on local-via allocation of P/G TSVs.

Our simulation results demonstrate that:

- Previous techniques cannot guarantee the EM reliability in 3D PDNs with via-first TSVs since they ignored the existence of local vias in P/G TSVs. In contrast, the proposed technique can provide more reliable and efficient design of 3D PDNs.
- The proposed technique has a superior timing performance in terms of worst slack compared to a routability-unaware solution.

The rest of this chapter is organized as follows. Preliminaries and related prior work are presented in Section 4.1. Section 4.2 describes the proposed local-via allocation

methodology in detail. In Section 4.3, we report simulation results. Finally, conclusions are drawn in Section 4.4.

4.1 Preliminaries and Related Work

4.1.1 Via-first TSV-based 3D power delivery network design challenges

The typical PDN of 3D ICs with via-first TSVs is presented in Figure 4.1. Unlike via-last TSVs which pierce through all the metal layers, via-first TSVs do not penetrate any metal layers. Instead, each TSV has two landing pads on M_{top} and M_{bottom}, and thus only affects the device layer. However, since the 2D power/ground mesh is located in the global and intermediate metal layers [126], the via-first P/G TSV needs stacked local vias to connect it with the landing pad on M_{bottom} [95]. Therefore, the via-first P/G TSVs affect metal layer as well as device layer in a similar way as via-last TSVs. Consequently, they function as both placement and routing obstacles, causing severe routing congestions if excessive local vias are allocated.

4.1.2 Electromigration reliability of TSVs

EM is a wear-out failure mechanism for metal interconnects, which is often caused by interconnect voiding from metal atomic diffusion. Generally, EM is explained as a two-phase process: i) void nucleation, followed by ii) void growth [125]. The two phases are dominated by different factors in 3D ICs: On the one hand, TSV-induced mechanical stress has a dominant influence on the void nucleation phase [127]; on the other hand, the effect of current density is the most dominant factor of the void growth [126]. However, for nanoscale copper-based interconnects, it is reasonable to assume a very short void nucleation time since it is nearly impossible to have void-free adhesion between copper and barrier/line material during the manufacturing process [128]. Moreover, as the intrinsic stress from TSV manufacturing in 3D ICs can shorten the nucleation time even further [127], the total long-term EM-related failure time is largely dominated by void growth rather than void nucleation, and hence the effect of stress can be ignored [126].

4.1.3 Electromigration-induced resistance increase of via-first TSV

In 3D ICs, P/G TSVs are prone to EM-induced voiding effects, which increase their resistances. Although the increase of resistance may not result in an immediate breakdown of 3D ICs, it can cause excessive IR-drop over time and degrade timing performance of circuits. Specific to via-first P/G TSVs, since it is necessary to use local vias for them to provide vertical power delivery, the calculation of their resistances should consider the existence of local vias.

In [126], an analytical resistance model based a resistance network was proposed. As illustrated in Figure 4.2, local vias are represented as a parallel resistance network since they are placed on the same TSV landing pad. Therefore, the equivalent resistance of a

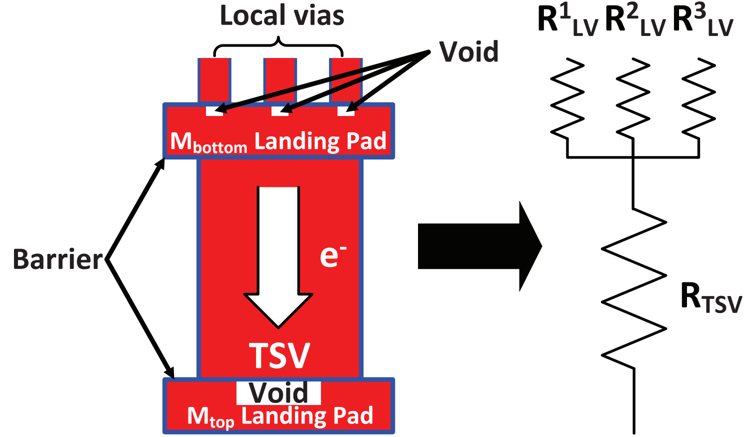


Figure 4.2: Illustration of a via-first P/G TSV with local vias and its equivalent resistance network.

via-first P/G TSV with n local vias, $R_{\mathrm{TSV}}^{\mathrm{Eq}}$, can be calculated in the following way:

$$R_{\mathrm{TSV}}^{\mathrm{Eq}} = R_{\mathrm{TSV}}(t) + \frac{1}{\sum_{i=1}^{n} 1/R_{\mathrm{LV}}^{i}(t)} \tag{4.1}$$

here $R_{\mathrm{TSV}}(t)$ is the resistance of TSV, and $R_{\mathrm{LV}}^{i}(t)$ is the resistance of the ith local via.

For $R_{\mathrm{TSV}}(t)$, it can be computed as in [124]:

$$R_{\mathrm{TSV}}(t) - R_{\mathrm{TSV}}(0) = \begin{cases} \frac{\rho_b}{4\pi t_{b,\mathrm{TSV}}} \ln\left(\frac{t}{t_0}\right) & t \geq t_0 \\ \frac{\rho_b t_{b,\mathrm{TSV}}}{\pi r_{\mathrm{TSV}}^2} \left[\frac{(t/t_0)}{1-(t/t_0)}\right] & t < t_0 \end{cases} \tag{4.2}$$

where $R_{\mathrm{TSV}}(0)$ is the TSV resistance at $t = 0$. t_0 defines as the time when the EM-induced void becomes equal to TSV radius, which can be expressed as:

$$t_0 = \frac{h_{\mathrm{TSV}} \pi r_{\mathrm{TSV}}^2 kT}{\alpha f \Omega A_l D_0 e Z^* \rho_{\mathrm{Cu}} j_{\mathrm{TSV}}} \exp\left(\frac{E_a}{kT}\right) \tag{4.3}$$

where j_{TSV} is the current density within the TSV. Other parameters in Equations (4.2) and (4.3) are listed in Table 4.1, and their values can be found in [126].

Since the reason behind EM-induced void nucleation and growth in local vias is the same as that of P/G TSVs [126], the resistance of a single local via $R_{\mathrm{LV}}(t)$ can be also calculated using Equations (4.2) and (4.3), but with different parameters (the values can be found in [126]). However, due to the non-uniform current distribution in the array of multiple local vias, its EM behavior is quite different from a single via [129]. The detailed solution for this issue will be presented in Section 4.2.3.

4.1.4 Related prior works

In conventional 2D ICs, the insertion of redundant vias can be applied as part of the routing step [130, 131] or as an additional post-layout optimization [132–134] in order

Table 4.1: Parameter description in Equations (4.2) and (4.3).

Parameters	Description
k	Boltzman const.
T	Temperature
α	Ratio of captured vacancies
f	Ratio of vacancy volume
Ω	Atomic volume
D_0	Initial diffusivity
e	Electron charge
Z^*	Effective charge const.
E_a	Activation energy
A_l	Vacancy absorbing area
ρ_{Cu}	Cu resistivity
ρ_{b}	Barrier resistivity
r_{TSV}	TSV radius
h_{TSV}	TSV height
$t_{b,\mathrm{TSV}}$	Barrier layer thickness at the bottom of TSV

to improve yield and reliability. However, it is necessary to consider 3D PDN design and optimization at early design stages to prevent costly redesign due to power delivery issues [103]. As essential components of 3D PDNs, the local vias inside P/G TSVs should also be inserted as early as possible. Therefore, here the conventional redundant via insertion techniques are infeasible since they insert the redundant vias too late in the design stages.

In 3D ICs, interest in EM reliability of P/G TSVs has surged in recent years, and several methods have been proposed to address this issue. In [121], a transient model of EM in P/G TSVs and TSV-to-wire interfaces in 3D PDN was proposed. However, since TSVs were assumed to connect directly to 2D P/G meshes without using local vias, such local via-unaware technique is only applicable to via-last TSVs. In [126], a comprehensive investigation on EM issue of via-first P/G TSVs was presented, in which the interplay between TSVs and local vias was considered. Nevertheless, the P/G TSV-induced routing congestion was ignored in this routability-unaware methodology. Although the general objective of 3D PDN optimization is to minimize the usage of P/G TSVs for alleviating the routing congestion [95], here it is the local vias in the P/G TSVs indeed that consume the routing area. Therefore, the number of local vias should be reduced for routability improvement, which in turn increases the current density of each via and may lead to shorter EM-induced lifetime of TSV. To handle the trade-off between EM reliability of P/G TSVs and the routability in the design of 3D ICs, it is imperative to determine the optimal number of local vias in each P/G TSVs, which is referred to as "EM-aware local-via allocation" problem in this work.

4.2 Proposed Electromigration-aware Local-via Allocation

4.2.1 Problem statement

In this work, an ILP formulation is used to solve this EM-aware local-via allocation problem. The formal problem statement is as follows:

- **Input**: i) A 3D IC placement consisting of a set of placed P/G TSVs; ii) a set of local-via allocation candidates; iii) the acceptable limit of TSV EM-induced average and worst IR-drop increase of P/G TSV network; iv) the target lifetime of the 3D PDN.

- **Output**: An optimal local-via allocation solution, which indicates that which local-via allocation candidate is selected for each P/G TSV.

- **Constraint**: EM-induced IR-drop increase of P/G TSV network (both the average and worst cases) should be within the given acceptable limit during the target lifetime of the 3D PDN.

- **Objective**: Minimize the routing congestion introduced by the insertion of local vias in each P/G TSV.

Before presenting the ILP formulation in detail, we clarify the assumption made for local-via allocation and introduce our notation used in this ILP formulation.

4.2.2 Assumption and notation

Assumptions

We make the following assumptions in order to simplify the optimization problem:

- The proposed local-via allocation is applied after 3D placement phase but prior to the global routing stage of 3D design. Therefore, the information of the assigned metal layer for each net, which can be obtained during the layer assignment stage, is unavailable during the local-via allocation. In this work, we always consider the worst case when estimating the routing congestion in order to guarantee the feasibility of the generated local-via allocations after the following layer assignment and detailed routing.

- Since local-via-induced routing congestion can be similar regardless of the particular metal layer used for the landing pads of TSVs [126], we assume that the landing pad is always located on M_{bottom} (i.e., M_1), and thus only local vias act as routing blockages. Moreover, the 2D PDN in each die is designed as a hierarchical structure, which consists of: i) global PDN, and ii) local PDN. Note that, here we made the same assumption as in [90, 108, 135, 136], and ignored the intermediate PDN in this work. The global and local PDNs are structured as a multilayer mesh grids, and designed in different levels of metal layers. As shown in Figure 4.3, the 2D

global PDN in each die is assumed to be located in M_7-M_{10} for 45 nm technology node [126], while the local one is located in M_1-M_2. In particular, since M_1 and M_2 might also be utilized for routing within the cells [137], here we assume that only M_3-M_6 are used for signal routing, without loss of generality. Therefore, the excessive local vias in each TSV could occupy significant area in the routing layers (i.e., M_3-M_6), which results in severe routing congestion. Please note that, here the benefit of the proposed approach in this work is not replied on these assumptions and the experimental setup.

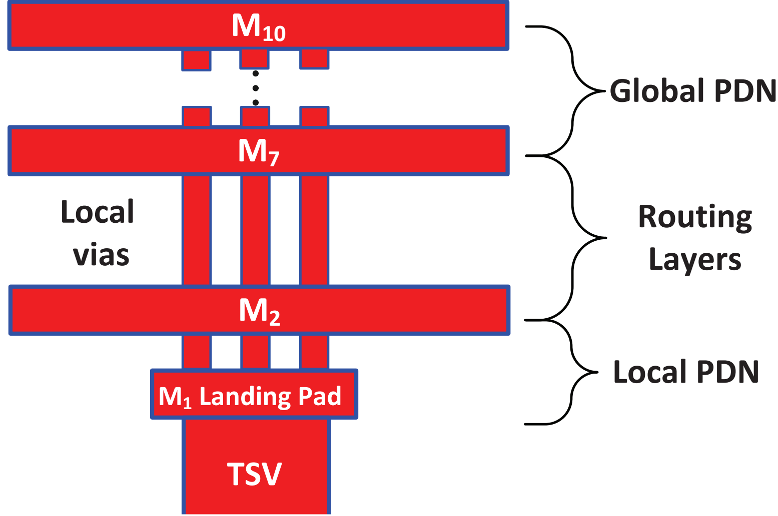


Figure 4.3: Illustration of a via-first P/G TSV with local vias and its corresponding impact on signal routing.

Notations

We use the following notations to facilitate the discussion.

- **S**: the set of P/G TSVs in the 3D IC;
- s_i: a P/G TSV in **S** (i.e., $s_i \in \mathbf{S}$);
- **C**: the candidate set of local-via allocations for each P/G TSV;
- c_j: a local-via allocation candidate in **C** (i.e., $c_j \in \mathbf{C}$);
- x_{ij}: a binary variable indicating whether s_i adopts c_j: $x_{ij} = 1$ when s_i adopts c_j, and $x_{ij} = 0$ otherwise;
- $\mathbf{LV}_{ij}$: the set of local vias of s_i when it adopts c_j;
- $R_{ij}(t)$: the resistance of s_i adopting c_j at time step t;

- **GC**: the set of grid cells partitioned the entire routing space;
- gc_k: a grid cell in **G** (i.e., $\mathrm{gc}_k \in \mathbf{GC}$);
- OVL_k: the track overflow of gc_k;
- $\mathbf{G}_{3\mathrm{D}}(t)$: the conductance matrix of the 3D PDN at time step t;
- $\mathbf{G}_{2\mathrm{D}}^{d}$: the conductance matrix of the 2D PDN of the dth die;
- $\mathbf{G}_{\mathrm{TSV}}(t)$: the conductance matrix of TSVs at time step t;
- $\mathbf{V}(t)$: the voltage matrix of the 3D PDN at time step t;
- **N**: the set of 3D PDN nodes;
- n: a 3D PDN node in **N** (i.e., $n \in \mathbf{N}$);
- $V_n(t)$: the voltage of n at time step t (i.e., $V_n(t) \in \mathbf{V}(t)$);
- **I**: the current vector of the 3D PDN;
- $\Delta V_{s_i}(t)$: the IR-drop of s_i after time step t;
- V_{dd}: the supply voltage of the 3D IC;
- $\Delta V_{dd,\min}^{\mathrm{avg}}$: the acceptable limit of EM-induced average IR-drop increase of the TSV network;
- $\Delta V_{dd,\min}^{\mathrm{worst}}$: the acceptable limit of EM-induced worst IR-drop increase of the TSV network;
- t_{LT}: the target lifetime of the 3D PDN;

4.2.3 Integer linear programming formulation

In this subsection, we describe the ILP model for minimizing the local-via-induced routing congestions under the constraints of the EM reliability in P/G TSVs, which is characterized by IR-drop increase.

Objective function

In order to gauge the severity of routing congestion, one of the metrics commonly used is the track overflow. After dividing the entire routing space into small subregions (referred to as “grid cells” in this chapter), the track overflow is used to measure the number of excess tracks required to route the wires in a grid cell [138]. For a grid cell gc_k, its track overflow OVL_k can be calculated as [137]:

$$\mathrm{OVL}_k = \max\left(0, \mathrm{RD}_k - \mathrm{RR}_k\right) \tag{4.4}$$

where RD_k and RR_k are the routing demand and resource on gc_k, respectively.

For RD_k, the most accurate value comes from global routing itself. However, since it is assumed that local-via allocation is applied prior to the routing phase (as mentioned in Section 4.2.2), a congestion estimation approach based on placement is desired in this scenario. In [137], a congestion estimator with good fidelity and high sensitivity to placement changes was proposed, which is applicable for our purpose. Therefore, we utilize the following approach to estimate the routing demand in this work: For each multi-pin net $\mathrm{net}_{k,u}$ in gc_k, its topology is determined using the two-bend LZ router developed in [139] after decomposing it into several two-pin connections. It was shown in [140] that the complexity to route a two pin net on a given g_x by g_y bin structure for a two pin net that spans x by y bins is $O(\log(x+y) \times \log(g_x+g_y))$. Afterwards, $\mathrm{RD}_{k,u}$, the routing demand for $\mathrm{net}_{k,u}$, is calculated as in [137]:

$$\mathrm{RD}_{k,u} = \begin{cases} w_{k,u}, & \text{if } \mathrm{net}_{k,u} \text{ crosses } \mathrm{gc}_k \\ 0.25 \times \mathrm{pin}_{k,u} \times w_{k,u} & \text{if } \mathrm{net}_{k,u} \text{ is within } \mathrm{gc}_k \\ 0 & \text{otherwise} \end{cases} \tag{4.5}$$

$$\mathrm{RD}_k = \sum_{\forall \mathrm{net}_{k,u} \text{ in } \mathrm{gc}_k} \mathrm{RD}_{k,u} \tag{4.6}$$

where $w_{k,u}$ is the wire width of $\mathrm{net}_{k,u}$, and $\mathrm{pin}_{k,u}$ is the number of pins in $\mathrm{net}_{k,u}$. Note that RD_k is a constant for a given placement.

For RR_k, it can be computed as in [137]:

$$\mathrm{RR}_k = \sum_{l=1}^{L} \mathrm{RR}_{k,l} \tag{4.7}$$

where L is the number of routing metal layers, and $\mathrm{RR}_{k,l}$ is the routing resource provided by the lth available routing metal layer. As suggested in [137], $\mathrm{RR}_{k,l}$ can be calculated as:

$$\mathrm{RR}_{k,l} = \frac{A_{k,l}^{\mathrm{AVBL}}}{w_l + s_l} \tag{4.8}$$

where w_l and s_l are width and space of metal wires in layer l, respectively. $A_{k,l}^{\mathrm{AVBL}}$ denotes the area of the lth metal layer available for routing over gc_k. Since only local vias can make routing blockage for via-first P/G TSVs [126], $A_{k,l}^{\mathrm{AVBL}}$ is the same for all the l routing layers, which can be calculated as:

$$A_{k,l}^{\mathrm{AVBL}} = A_k^{\mathrm{grid_cell}} - A_k^{\mathrm{local_vias}} \tag{4.9}$$

where $A_k^{\mathrm{grid_cell}}$ is the area of gc_k, and $A_k^{\mathrm{local_vias}}$ is the overlapped area between gc_k and all the local vias. Therefore, with the different local-via allocation in each TSV, $A_k^{\mathrm{local_vias}}$ is varied, and can be represented as:

$$A_k^{\mathrm{local_vias}} = \sum_{i}^{|\mathbf{S}|} \left(\sum_{j}^{|\mathbf{C}|} \mathrm{overlapped}\left(\mathbf{LV}_{ij},\ \mathrm{gc}_k\right) \cdot x_{ij} \right) \tag{4.10}$$

here overlapped $(\mathbf{LV}_{ij},\ \mathrm{gc}_k)$ is the area of overlapped region between $\mathbf{LV}_{ij}$ and gc_k. Then RR_k can be expressed as a linear combination of the binary variables $\{x_{ij}\}$.

Therefore, combining Equations (4.4), (4.5), (4.6), (4.7), (4.8), (4.9), and (4.10), we state the objective function of the ILP formulation as follows:

$$\text{Minimize:} \quad \sum_{k=1}^{|\mathbf{GC}|} \mathrm{OVL}_k \tag{4.11}$$

The above minmax objective can be simply linearized using the following standard technique associated with ILP modeling:

$$\begin{aligned} \forall k: \quad & y_k \geq 0, \quad y_k \geq \mathrm{RD}_k - \mathrm{RR}_k \\ \text{Minimize:} \quad & \sum_{k=1}^{|\mathbf{GC}|} y_k \end{aligned} \tag{4.12}$$

Constraints

The constraints of the ILP formulation can be divided into two categories:

- Local-via number constraints:

$$\forall i: \quad \sum_{j=1}^{|\mathbf{C}|} x_{ij} = 1 \tag{4.13}$$

- TSV EM-induced IR-drop increase constraints:

$$\frac{\sum_{s_i \in \mathbf{S}} \Delta V_{s_i}(t_{LT})}{|\mathbf{S}|\, V_{dd}} \geq \Delta V_{dd,\min}^{\mathrm{avg}} \tag{4.14}$$

$$\frac{\min\left\{\Delta V_{s_1}(t_{LT}) \ldots \Delta V_{s_{|\mathbf{S}|}}(t_{LT})\right\}}{V_{dd}} \geq \Delta V_{dd,\min}^{\mathrm{worst}} \tag{4.15}$$

The set of constraints given by (4.13) implies that for each P/G TSV $s_i \in \mathbf{S}$, it can only adopt one candidate in $\mathbf{C}$. The constraints (4.14) and (4.15) ensure that the EM-induced IR-drop (both the average and worst cases) of TSV network should be still within the given acceptable limit after t_{LT}.

In order to obtain $V_n(t)$ in the constraints (4.14) and (4.15), the following linear system need to be solved:

$$\mathbf{G}_{3\mathrm{D}}\mathbf{V} = \mathbf{I} \tag{4.16}$$

Here, a DC resistive model of 3D PDN is assumed, in which the on-chip PDN is modeled as a resistive grid and $\mathbf{I}$ is time-invariant. The reasons for such assumption are as follows: i) Typically, the on-chip PDN can be modeled as a conductive grid with resistive, inductive and capacitive parasitic elements (i.e., so-called “Resistive-Capacitive-Inductive”

model). Although the current distribution also depends on the capacitive and inductive elements, these ones have a low impact in comparison with the resistive ones, which is indeed the dominant on-chip parasitic element of the PDN. Therefore, the capacitive and inductive elements of the 3D PDN are not considered in this work, and the PDN of each die in a 3D IC is modeled as a two-dimensional resistive grid, which is connected by P/G TSVs. ii) Since EM is a long-term cumulative failure mechanism, the current changes in 3D PDN during normal operation are not significant. Although the dynamic analysis method is claimed to provide more precise insight into the behaviour of the PDN, the DC analysis can model the long-term behaviour of the PDN by averaging the tap currents, which is accurate-enough for evaluating the EM-induced IR-drop. In this scenario, an effective-EM current is desirable [141]. Since P/G TSVs mostly carry uni-directional currents, this effective-EM current can be derived from DC EM analysis based on time-average current density [142]. After the computation of an average power value for each logic gate, these power values are converted to DC current sources and are attached to each 2D P/G mesh in the 3D PDN at the appropriate PDN nodes per gate. Note that, since the proposed method is generic, it is reasonable to extend it by taking more complicated and more accurate PDN models into account.

In Equation (4.16), $\mathbf{G}_{\text{3D}}$ can be constructed using $\mathbf{G}_{\text{2D}}^{d}$ and $\mathbf{G}_{\text{TSV}}$ [104]. For a two-die 3D IC, it can be expressed as follows:

$$\mathbf{G}_{\text{3D}} = \begin{pmatrix} \mathbf{G}_{\text{2D}}^{1} + \mathbf{G}_{\text{TSV}} & -\mathbf{G}_{\text{TSV}} \\ -\mathbf{G}_{\text{TSV}} & \mathbf{G}_{\text{2D}}^{2} + \mathbf{G}_{\text{TSV}} \end{pmatrix} \in \mathbb{R}^{|\mathbf{N}|\times|\mathbf{N}|} \tag{4.17}$$

This can easily be extended for a multiple-die 3D IC by applying this equation in a pair-wise manner between adjacent dies [104]. Due to EM degradation, $\mathbf{G}_{\text{TSV}}$ should be time-varying, and can be represented as:

$$\mathbf{G}_{\text{TSV}}(t) = diag(g_1(t), ..., g_i(t), ..., g_{|\mathbf{S}|}(t)) \tag{4.18}$$

here $g_i(t)$ is the conductance of s_i at time step t. By considering the various local-via allocation candidates, $g_i(t)$ is written as:

$$g_i(t) = \sum_{j=1}^{|\mathbf{C}|} \frac{x_{ij}}{R_{ij}(t)} \tag{4.19}$$

Hence, $g_i(t)$ can be expressed as a linear combination of the binary variables $\{x_{ij}\}$, and thus $\mathbf{G}_{\text{3D}}$ is also a function of $\{x_{ij}\}$.

$R_{ij}(t)$ can be calculated using Equation (4.1) after obtaining the resistance of TSV $R_{\text{TSV}}(t)$ and the resistance of the allocated local-via array $R_{\text{LV}}(t)$. Regarding $R_{\text{TSV}}(t)$, it can be computed following Equation (4.2). Note that, since the power grid current distribution varies according to circuit working mode or transistor activity factors, here we consider effective (i.e., long term average) current density for each TSV in Equation (4.3) in order to evaluate the overall EM reliability. In [143], a method was proposed to obtain such effective current, and we assume that the effective current is already provided.

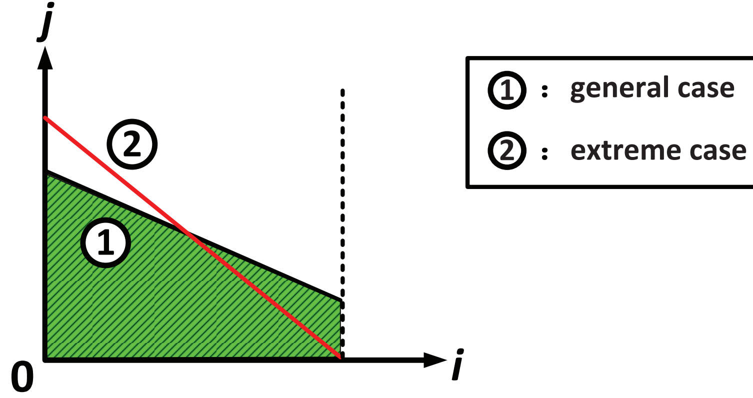


Figure 4.4: Non-uniform distribution of current density within each local via in the array. Here i is the index of each via, and j is its current density.

For calculating $R_{\mathrm{LV}}(t)$, it is necessary to obtain the current density through each local via in advance. According to the experimental results in [129], the EM behavior of local-via array is quite different from a single via due to the non-uniform current distribution within the array. However, due to resistive effect, the uneven current distribution in a via-array is linear [144]. Figure 4.4 illustrates a general uneven current distribution where some vias carry less current, but others carry more (as shown in the curve ①). As in [144], here we consider an extreme triangular distribution (i.e., the red line ② in Figure 4.4), in which vias at one end carry zero current density, and on the other end, vias carry current the density of $2j_{\mathrm{avg}}$. Here j_{avg} is the via-array average current density. Note that, here the two distributions have the same amount of total current, i.e., the area below each curve is the same. In this way, the current density within each via can be calculated for each allocation candidate, and accordingly the resistance of a via-array.

Due to the existence of binary variables in $\mathbf{G}_{3\mathrm{D}}$, it is very time-consuming to solve the linear system (4.16) directly. Instead, we take each linear equation in this system as a constraint, and add them into the ILP formulation. In order to linearize the bilinear terms in these constraints, we replace them by McCormick envelope [111]. More details can be found in [145].

Therefore, combining Equations (4.12), (4.13), (4.14), (4.15) and (4.16), we obtain an ILP formulation to minimize the local via-induced routing congestion in each P/G TSV with EM reliability consideration. This optimization problem can be solved by an ILP solver.

4.2.4 Scalability analysis

Scalability has always been a critical issue for ILP-based methods. The size of the formulation, which includes the number of variables and constraints, reflects the complexity of ILP formulation. As explained below, our approach offers a very good scaling behavior for circuits with increasing gate count.

Suppose we consider a 2D circuit with the number of gate N_{gate}, and partition it

into an L-layer 3D IC uniformly. Therefore, assuming that each PDN node in 2D power mesh can drive g logic gates at most, the number of PDN nodes in each layer, $|\mathbf{N}|_{2d}$, is equal to

$$|\mathbf{N}|_{2d} = \lceil \frac{N_{gate}}{L \times g} \rceil \tag{4.20}$$

Therefore, even though we place P/G TSVs at each PDN node, the number of TSVs, $|\mathbf{S}|$, is

$$|\mathbf{S}| = (L-1)\,|\mathbf{N}|_{2d} \tag{4.21}$$

Note that there is (are) $L-1$ layer(s) of P/G TSVs connecting L layers 2D power meshes in the 3D IC. Therefore, the number of the binary variables $\{x_{ij}\}$, N_{var}, in the ILP formulation is equal to

$$N_{var} = |\mathbf{C}|\,(L-1)\,|\mathbf{N}|_{2d} \tag{4.22}$$

here $|\mathbf{C}|$ is the number of local-via allocation options for each TSV. In addition, the number of total constraints (including the constraints for linearizing the bilinear terms in the ILP formulation) N_{constr} can be presented as

$$N_{constr} = 2\,|\mathbf{GC}| + [2\,|\mathbf{C}|\,(L-1) + 3L - 2]\,|\mathbf{N}|_{2d} + 1 \tag{4.23}$$

Therefore, for a fixed $\mathbf{GC}$, both the number of binary variables N_{var} and the number of constraints N_{constr} scale linearly with the number of gates. Thus, we can expect a good scalability of our ILP formulation for large circuits.

4.2.5 Discussion

Here we discuss the practicality of the propose method, we perform the following discussions from the perspectives of manufacturing and design complications. First, due to the huge size of P/G TSV compared to signal one, the number of P/G TSVs in a typical 3D chip is far limited compared to signal TSVs [95, 146, 147]. In addition, as a user-defined parameter, the number of local-via allocation candidates (i.e., the options of different local-via number in TSVs) can be tuned according to varied design requirements. Therefore, from the manufacturing perspective, it is totally reasonable to place different number of vias in each TSVs. Second, in order to finalize the optimization once during design-time as proposed previously, it is desirable to consider either representative or worse-case workload among all the applications. However, such assumption will not affect the practicality of the proposed approach.

4.3 Simulation Results

4.3.1 Simulation setup and implementation flow

For our simulations, eight 3D benchmark designs were used. Besides *des_perf-i*, *cf_rca_16-i*, and *cf_fft_256_8-i* (i $= 2, 4$) selected from OpenCore benchmark suite [148],

two artificial circuits *des_cf_fft-i* (i = 2, 4) were also used by combining *des_perf-i* and *cf_fft_256_8-i* together. Here, i is the number of stacked dies in each design. The detailed information about the benchmarks, including area, power, gate number, 3D PDN design and nets being considered for routing, etc., are listed in Table 5.1. Moreover, the relevant parameter values of P/G TSV are also specified in Table 4.3. The simulations were performed on a server with four AMD Opteron 6174 processors and 256 GB RAM.

The basic implementation flow is as follows. First, based on the given netlist of each die, Cadence SoC Encounter was used to perform placement for all the dies in each design separately using the Nangate 45 nm library [115]. In the floorplan, P/G and signal TSVs were placed regularly. In this experiment, we assume that all the *Controlled Collapse Chip Connection* (C4) bumps are aligned with the TSV locations to minimize IR-drop loss. As a result, a current crowding analysis between TSVs and C4 bumps was not necessary in this work. Then, using a global router "Labyrinth" [139], the routing demand of each grid cell was estimated, and the routing resource consumed by local vias in each P/G TSV can be calculated according to the given allocation candidates. In this work, 7 candidates were considered, in which the local vias are placed regularly on TSV landing pad with the different number 16, 64, 144, 256, 400, 576, and 676 [3], respectively [126]. Based on the calculated routing demand of placement and routing resource consumed by local vias, the objective function of the proposed ILP formulation can be constructed using Equation (5.24).

In order to obtain the constraints of the proposed ILP formulation, the major concern are: i) how to extract current vector $\mathbf{I}$ in Equation (4.16), and ii) how to obtain the thermal profile for each P/G TSV. For the first aspect, the power consumption of each PDN node need to be collected by adding the dynamic and leakage currents of all the logic gates connected to it. In order to obtain the leakage current, all standard cells in the library are characterized by accurate SPICE simulations, and this information is then stored in a LUT. For the calculation of dynamic current, we extracted signal probabilities and activity factors for all the internal nodes in the circuit based on a set of representative input vectors [110]. Then all the information was sent into a power analyzer to obtain $\mathbf{I}$. In order to obtain the thermal profile, the obtained power profile was forwarded to 3D Hotspot [149] for temperature estimation with the same configuration setting as in [149]. Note that, since we assume that EM is a long-term degradation effect, here a steady-state temperature analysis is sufficient. Afterwards, we can construct the constraints by Equation (4.16). The generated ILP formulation was solved by CPLEX [119] to determine the optimal local-via allocation for each TSV. In order to evaluate the local-via-induced routing congestion, *Wire-Length* (WL) and *Critical Path Delay* (CPD) were used as a performance metric in this work, which can be obtained during post-routing stage by P&R commercial tool [4].

[3] 676 is the maximum number of local vias that can be packed within TSV landing pad for 45 nm technology [126]

[4] Here we used Cadence SoC Encounter to obtain them for each design.

Table 4.2: Design Statistics for all benchmark.

Benchmark	Area (μm^2)	Power (mW)	# Gates	# FFs	# signal TSVs	# P/G TSVs	# C4	# Nets	Design utilization (%)
des_perf-2	65,025	63.5	24,267	1,984	369	81	25	26,662	37.52%
des_perf-4	38,025	64.7	24,267	1,984	1,220	147	49	26,317	36.57%
cf_rca_16-2	342,225	172.5	136,144	20,480	582	400	100	168,301	40.50%
cf_rca_16-4	119,025	175.2	136,144	20,480	1,451	432	144	168,001	59.78%
cf_fft_256_8-2	416,025	491.2	223,550	75,723	1,569	484	121	373,796	80.39%
cf_fft_256_8-4	275,625	496.1	223,550	75,723	2,100	972	324	373,796	61.37%
des_cf_fft-2	540,225	554.7	247,817	77,707	1,938	625	169	400,458	66.47%
des_cf_fft-4	342,225	560.8	247,817	77,707	3,320	1,200	400	400,113	53.54%

Table 4.3: Experimental setting for P/G TSV.

Item	
TSV radius (μm)	1.15 [125]
TSV height (μm)	15 [125]
TSV pitch (μm)	30
TSV landing pad size (μm $\times$ μm)	3.6 $\times$ 3.6 [126]
TSV landing pad thickness (μm)	0.13 [126]

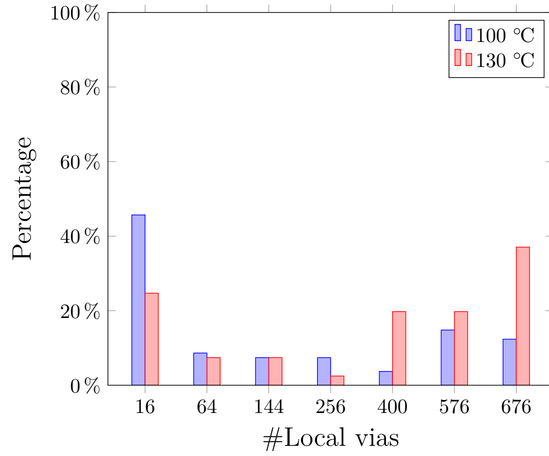
4.3.2 Impact of temperature on local-via allocation

In this subsection, the impact of temperature on local-via allocation is studied. According to Equations (4.2) and (4.3), higher temperature can aggravate EM-induced IR-drop due to the increased TSV resistance R_{TSV}. Therefore, in order to satisfy the constraints of EM reliability after a long-term operation, it is desirable to place more local vias for each TSV under higher temperature, which leads to more severe routing congestion and larger WL. Since temperature is in the exponent of diffusivity in Equation (4.3), the EM effects worsens exponentially with increased temperature.

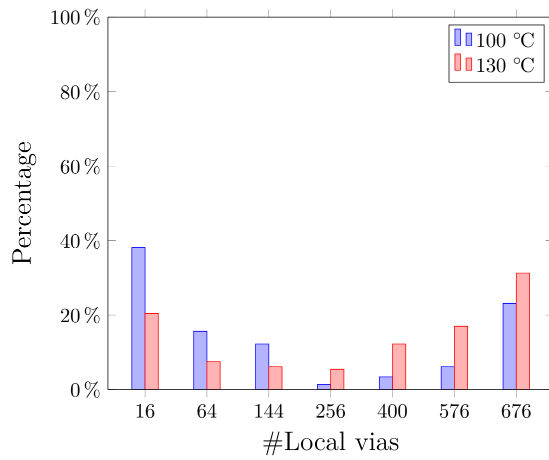
Here we performed the proposed approach with $t_{\text{LT}} = 3.2$ yrs and $\Delta V_{dd,\min}^{\text{avg}} = \Delta V_{dd,\min}^{\text{worst}} = 0.9$ on the benchmarks *des_perf-2* and *des_perf-4* under two different worst case temperatures 100 ℃ and 130 ℃, respectively. By using the proposed optimization method, the minimized routing overflow and satisfied IR-drop values (including the worst and average cases) can be obtained, listed in Table 4.4. Due to the higher IR-drop with the increased temperature, it is necessary to allocate more local vias for each P/G TSV. According to our simulation results, for the two benchmarks, the average number of local vias per TSV is increased by 74.34% and 60.74%, respectively. The detailed allocations are also presented in Figure 4.5. Consequently, it is necessary to reduce operating temperature in 3D ICs for EM reliability improvement and efficient design cost.

4.3.3 Impact of circuit partitioning on local-via allocation

In this work, we consider the impact of stacking by implementing the benchmarks into 2- and 4-layers. The major issue related to an implementation with more layers is heat dissipation. Stacked dies result in higher power density, yet the underfill between adjacent layers generally has very low thermal conductivity, making vertical heat dissipation a severe problem. Therefore, the temperature variation, especially the inter-die one, becomes more significant with the increased number of stacked dies in 3D ICs. This subsequently leads to more dramatic resistance increase on P/G TSVs. Note that, although power density could be decreased by partitioning the same design into more layer, the reduced current densities within TSVs exhibit weaker dependency on EM degradation compared to the higher temperature, according to Equation (4.3). Therefore, it is more challenging to deliver enough current to all parts of the 3D stack while guaranteeing high power integrity for the stable operation of 3D ICs.



(a) *des_perf-2*



(b) *des_perf-4*

Figure 4.5: Local-via distribution of P/G TSVs in *des_perf-2* and *des_perf-4* under worst case 100 °C and 130 °C temperatures.

Table 4.4: The impact of temperature on the local-via allocation in terms of optimized overflow and IR-drop ($t_{\text{LT}} = 3.2$ yrs and $\Delta V^{\text{avg}}_{dd,\min} = \Delta V^{\text{worst}}_{dd,\min} = 0.9$).

	Temperature (°C)	
des_perf-2	100	130
Overflow	0	0
Worst IR-drop (mV)	97.74	99.34
Average IR-drop (mV)	48.55	49.32
Average #Vias	226	394
	Temperature (°C)	
des_perf-4	100	130
Overflow	0	0
Worst IR-drop (mV)	99.76	99.91
Average IR-drop (mV)	55.93	57.16
Average #Vias	242	389

Here, we conducted the simulations on *des_perf-2* and *des_perf-4* with $t_{\text{LT}} = 6.4$ yrs, and applied the proposed approach to obtain the optimal local-via allocation for each TSV under the same constraints of EM reliability. As shown in Table 4.5, both the worst and average IR-drop are increased when partitioning the same design into more layers. Moreover, in order to satisfy the same constraints of EM reliability, it is necessary to place more local vias in each P/G TSV, which results in routing congestion. As shown, the routing overflow of *des_perf-4* can not be reduced to 0, even after the optimization of local-via allocation. The reason is that the starting points of overflow in the two benchmarks are different: when inserting the local vias with maximum number (i.e., 676 in our experimental setting) in each P/G TSV, the total overflow of *des_perf-2* is 2104.78, while the total overflow of *des_perf-4* is 4973.24. In addition, the detailed local-via allocations for both the two benchmarks are presented in Figure 4.6. According to our simulation results, the average number of local vias per TSV is increased by 79.74% when partitioning the 2D design from 2 layers into 4 layers.

Table 4.5: The impact of circuit partition on the local-via allocation in terms of optimized overflow and IR-drop ($t_{\text{LT}} = 6.4$ yrs and $\Delta V^{\text{avg}}_{dd,\min} = \Delta V^{\text{worst}}_{dd,\min} = 0.9$).

	Partitioning	
	2-die	4-die
Overflow	0	4400.42
Worst IR-drop (mV)	99.19	99.87
Average IR-drop (mV)	49.25	59.73
Average #Vias	311	559

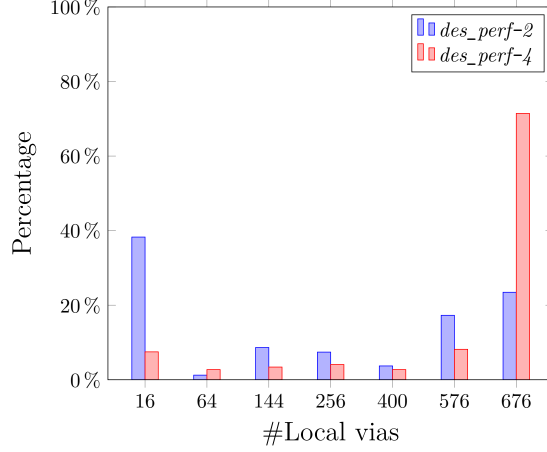


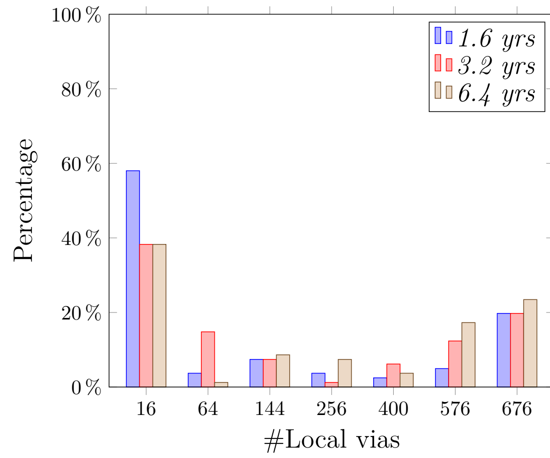
Figure 4.6: Local-via distribution of the P/G TSVs in *des_perf-2* and *des_perf-4* under the same temperature.

4.3.4 Impact of target lifetime on local-via allocation

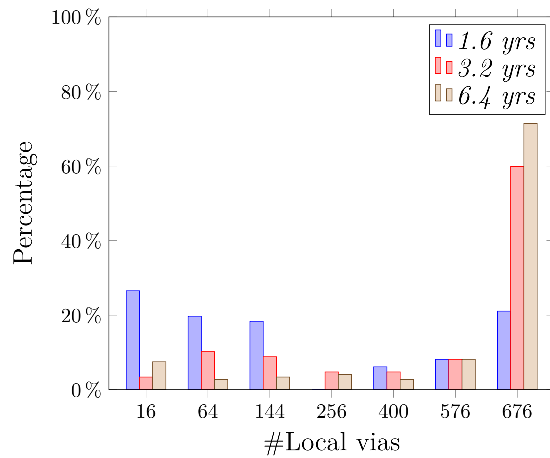
The target lifetime is another important metric in the proposed approach. In this subsection, we consider three different target lifetimes of 3D ICs, and evaluate its impact on the local-via allocation. For the benchmarks *des_perf-2* and *des_perf-4*, we performed the proposed technique to obtain the optimal local-via allocation with the constraints of $\Delta V_{dd,\min}^{\text{avg}} = \Delta V_{dd,\min}^{\text{worst}} = 0.9$. For the different target lifetimes, we report the achieved worst and average IR-drop, and also the optimal overflow values in Table 4.6. Besides, the generated allocation for each target lifetime is also illustrated in Figure 4.7. According to our simulation results, it is desirable to allocate more local vias for achieving a longer lifetime, which in turn leads to congestion problem during routing stage.

Table 4.6: The impact of target lifetimes on the local-via allocation in terms of optimized overflow and IR-drop ($\Delta V_{dd,\min}^{\text{avg}} = \Delta V_{dd,\min}^{\text{worst}} = 0.9$).

	Lifetime (yrs)		
des_perf-2	1.6	3.2	6.4
Overflow	0	0	0
Worst IR-drop (mV)	97.41	97.70	99.19
Average IR-drop (mV)	48.45	48.54	49.25
Average #Vias	203	258	311
	Lifetime (yrs)		
des_perf-4	1.6	3.2	6.4
Overflow	0	3709.72	4400.42
Worst IR-drop (mV)	97.32	97.92	99.87
Average IR-drop (mV)	50.32	55.38	59.73
Average #Vias	257	502	559



(a) *des_perf-2*



(b) *des_perf-4*

Figure 4.7: The detailed local-via allocations for *des_perf-2* and *des_perf-4* under the different constraints of target lifetimes.

4.3.5 Comparison with prior work

As summarized in Section 6.1.2, prior work can be divided into two categories: local via-unaware technique [121] and routability-unaware technique [126]. Here we compare our proposed methodology with the two techniques, respectively.

Comparison with local via-unaware technique

In [121], P/G TSVs were assumed to connect directly to 2D P/G meshes without using local vias. This local via-unaware technique is not applicable to the 3D ICs with via-first P/G TSVs due to the following drawbacks:

- The impact of local vias on IR-drop was ignored, resulting in inaccurate estimation and unreliable 3D PDN design.
- Instead of local vias, the entire TSV is treated as routing blockage, which overestimates the TSV-induced routing congestion and results in unnecessary over-design.

Here the evaluation of the proposed technique was performed on all the benchmarks with $t_{\mathrm{LT}} = 3.2$ yrs and $\Delta V^{\mathrm{avg}}_{dd,\min} = \Delta V^{\mathrm{worst}}_{dd,\min} = 0.9$. The average and worst IR-drop of 3D PDN were calculated using the two different techniques, respectively. Moreover, the WL and CPD of each design can be obtained based on the placed routing blockages (i.e., local vias for the proposed approach and P/G TSVs for local via-unaware method). According to the results in Table 6.4, local via-unaware solution overestimates the WL since treating the entire TSV as a routing blockage. Even worse, such inaccurate estimation leads to the error of post-routing timing analysis. For example, for benchmark *cf_fft_256_8-2*, since the via-unaware technique predicates the CPD pessimistically (from 3.341 ns to 4.3 ns), it has to add unnecessary timing margin for the critical paths, resulting in the degradation of timing performance. According to our simulation results, such error of CPD estimation can be up to 22.3%. In addition, as the resistances attributed by local vias are totally ignored, the via-unaware technique underestimates the IR-drop (both the worst and average cases) significantly, and generates an unreliable 3D PDN design.

Comparison with routability-unaware technique

In [126], since the analysis and optimization of via-induced routing congestion was ignored completely, each TSV has to adopt the allocation candidate with the same number of local vias (e.g., 676 local vias per TSV in [126]), in order to guarantee the long-term EM reliability of 3D PDNs. However, such coarse allocation solution could lead to unnecessary over-design, which results in exacerbated routing congestion and timing performance degradation. By contrast, the proposed technique can allocate appropriate number of local vias for each TSV according to its location.

In order to highlight its superiority, the proposed technique was implemented on all benchmarks with $t_{\mathrm{LT}} = 3.2$ yrs and $\Delta V^{\mathrm{avg}}_{dd,\min} = \Delta V^{\mathrm{worst}}_{dd,\min} = 0.9$. After generating the optimal local-via allocations, the WL and CPD of each design can be obtained, and compared to the baseline results. Here we conducted the baseline as follows: As

Table 4.7: Comparison between the proposed technique and the local vias-unaware technique [121].

Benchmark	WL (μm)			CPD (ns)			IR-drop (mV)			
							Average		Worst	
	Baseline [7]	Proposed	Improvement (%)	Baseline [7]	Proposed	Improvement (%)	Baseline [7]	Proposed	Baseline [7]	Proposed
des_perf-2	429,494.21	424,917.81	1.1	1.898	1.874	1.3	25.81	39.29	52.18	79.13
des_perf-4	453,797.14	451,182.73	0.6	2.181	2.022	7.3	35.52	55.38	61.69	97.32
cf_rca_16-2	2,009,010.09	2,004,749.03	0.2	3.069	2.910	5.2	22.75	49.01	45.87	98.45
cf_rca_16-4	1,920,257.31	1,914,771.52	0.3	3.465	2.770	20.1	35.73	53.08	64.25	100
cf_fft_256_8-2	5,024,752.20	4,979,298.16	0.9	4.300	3.341	22.3	19.98	47.18	47.18	99.14
cf_fft_256_8-4	6,087,124.05	6,003,182.60	1.4	4.679	4.011	14.3	30.67	61.23	49.35	99.71
des_cf_fft-2	5,753,217.92	5,679,116.47	1.3	4.721	3.715	21.3	40.11	59.80	52.65	96.33
des_cf_fft-4	6,541,132.73	6,409,001.84	2.0	5.292	4.671	11.7	45.67	63.22	58.56	99.04

in [126], we assumed that each P/G TSV adopts the same local-via allocation. Among all the candidates, we selected the feasible one (i.e., satisfying the IR-drop constraints) resulting in the minimum CPD as the baseline. According to our simulation results listed in Table 6.5, although both of them can satisfy the IR-drop constraints, it is unavoidable to set a larger margin in the baseline case in order to satisfy the timing constraints, which leads to performance degradation. On the contrary, our proposed technique has a better timing performance in terms of CPD value with the refined local-via allocation of each TSV. Please note that, compared to the baseline allocation with the maximum local-via count (i.e., 676 local vias per TSV in our work), the optimized local-via allocation could result in IR-drop increase due to the reduction in the number of local vias for each TSV. However, such delay impact can be taken into account by a voltage-droop-aware timing analysis [150], in which the impact of IR-drop on CPD can be considered and optimized after the proposed local-via allocation in this work.

Table 4.8: Comparison between the proposed technique and the routability-unaware technique [126].

Benchmark	WL (μm)			CPD (ns)		
	Baseline [13]	Proposed	Improvement (%)	Baseline [13]	Proposed	Improvement (%)
des_perf-2	428,370.96	424,917.81	0.8	1.886	1.874	0.6
des_perf-4	453,797.14	451,182.73	0.6	2.181	2.002	8.2
cf_rca_16-2	2,006,340.18	2,004,749.03	0.8	2.942	2.910	1.1
cf_rca_16-4	1,919,625.52	1,914,771.52	0.3	3.057	2.770	9.4
cf_fft_256_8-2	5,024,152.66	4,979,298.16	0.9	3.612	3.341	7.5
cf_fft_256_8-4	6,069,416.91	6,003,182.60	1.1	4.354	4.011	7.9
des_cf_fft-2	5,731,943.08	5,679,116.47	0.9	4.272	3.715	13.0
des_cf_fft-4	6,507,954.29	6,409,001.84	1.5	4.951	4.671	5.7

4.3.6 Runtime analysis

To obtain the runtime for the proposed technique, the experiments were performed on all the six benchmarks. By varying t_{LT}, the average runtime for each benchmark was calculated. This measure consists of two parts: the runtime for estimating the routing congestion (i.e., t_{routing}) and the runtime for solving the ILP formulation (i.e., t_{solving}). As illustrated in Table 4.9, both of these steps can be finished within a few minutes, even for the largest design.

4.4 Conclusion of Chapter

EM-related TSV reliability is one of the key obstacles for industry adoption of 3D ICs. As EM-prone structure, P/G TSVs should be carefully designed to mitigate the adverse effect of EM. In addition, the impact of P/G TSVs on the routiability and timing performance of 3D ICs is also significant due to the considerable consumption of routing resource by local vias inside P/G TSVs. Therefore, it is imperative to consider the local via-induced routing congestion during 3D PDN design, which was ignored in prior work.

Table 4.9: The runtime of the proposed methodology for all benchmarks.

Benchmark	t_{routing}(s)	t_{solving}(s)	t_{total}(s)
des_perf-2	0.61	3.10	3.71
des_perf-4	1.34	100.80	102.14
cf_rca_16-2	11.06	202.87	213.93
cf_rca_16-4	10.98	407.44	418.42
cf_fft_256_8-2	16.13	372.70	388.83
cf_fft_256_8-4	13.52	414.56	428.08
des_cf_fft-2	19.11	391.74	410.85
des_cf_fft-4	23.12	457.89	481.01

To overcome this drawback, a novel methodology is proposed to handle the trade-off between EM reliability of P/G TSVs and the timing performance of 3D ICs. By determining the optimal local-via allocation for each P/G TSV, the local via-induced routing congestion can be minimized while satisfying the given requirement of EM reliability.

5 Defect Clustering-aware TSV Repair for Yield Enhancement

Concerns related to manufacturing yield constitute one of the key obstacles in the widespread industry adoption of 3D integration technology [151, 152]. In general, the yield of 3D ICs can be reduced due to the defects in stacked dies (i.e., stack yield loss) or defects that occur during the assembly process (i.e., assembly yield loss) [34]. In the former case, it is critical to conduct pre-bond testing to prevent the stacking of defective dies [153]. Several die/wafer matching and inter-die repair strategies have also been proposed to enhance stack yield [154–156]. For the latter case, the addition of *Spare Through-Silicon Via* (s-TSV) to repair defective *Functional Through-Silicon Via* (f-TSV) is an effective method for increasing yield and ensuring reliability.

A number of s-TSV allocation strategies have been proposed in the literature [157–163]; however, these methods only consider uniform TSV placement (i.e., TSVs are placed in a uniform grid on the die). Although such a uniform TSV layout offers advantages such as lower heat dissipation and stronger package bonding [164], non-uniform TSV placement allows more design flexibility and leads to shorter wirelength [165]. As a result, non-uniform TSV placement provides two important benefits, namely lower latency and power reduction. However, due to the added degree of freedom in the locations of f-TSVs associated with non-uniform placement, it is a challenge to enhance the yield for such designs, and advances in s-TSV allocation methods are needed to achieve the above performance benefits.

In order to address this problem, s-TSV allocation techniques have recently been proposed for non-uniform TSV placement [161, 162]. However, there still remain three major challenges that limit the practicality of these techniques:

- First, prior work assumes that TSV defects are uniformly distributed. However, faulty TSVs tend to be clustered, e.g., due to imperfect bonding [166, 167]. Clustering leads to the location-dependency of TSV defect probabilities [159]. In other words, the defect probability of each TSV is related to its location. Ignoring this location-dependency will result in inaccurate yield estimation and an unrealistic solution to the problem of s-TSV allocation. Therefore, it is imperative that we take defect clustering into account during s-TSV allocation.

- Second, prior work assumes that s-TSVs can be inserted anywhere on the chip, without considering whitespace constraints. However, ignoring the whitespace constraint may result in overlap between f-TSVs and the s-TSVs after s-TSV allocation [162], and post-processing has to be carried out by relocating the f-TSVs [162]. Then, due to location dependency, the defect probability of the relocated f-TSV

will change after the allocation of the s-TSVs, which can result in a situation where repair is not feasible. Therefore, the whitespace constraint must be taken into account during s-TSV allocation.

- Finally, all prior work on s-TSV allocation only considers the delay overhead due to signal re-routing. However, TSV-induced-stress can also cause significant delay changes on critical paths, which can be up to 8.9% [100]. In order to avoid timing violations when the signal is re-routed, it is essential to take the TSV-stress-induced delay overhead into account during s-TSV allocation.

In this chapter, an efficient s-TSV allocation technique is proposed to overcome the above limitations. The proposed technique consists of two stages:

- Global partitioning: We partition the f-TSVs into multiple groups with a flexible number of f-TSVs and assign one or more s-TSVs to each group to minimize the hardware cost while satisfying the target chip yield.
- Detailed insertion of s-TSVs: We determine the exact location of each assigned s-TSV to minimize the delay overhead introduced by signal re-routing and TSV-induced stress.

Overall, the contributions of this work are summarized as follows:

- In order to obtain a cost-effective allocation solution in terms of s-TSV count, a greedy group merging algorithm is proposed to solve the global partitioning problem under the constraints of delay overhead. We have also proven that this optimization problem is $\mathcal{NP}$-hard.
- An ILP-based technique is developed to determine: i) the appropriate location candidates for s-TSV insertion, and ii) the optimal mapping between inserted s-TSVs and assigned f-TSVs for minimizing the delay overhead introduced by signal re-routing and TSV-induced stress.
- We conduct more realistic evaluations of yield enhancement, considering various impacts on s-TSV allocation, including defect-clustering effect, non-uniform temperature distribution, and timing criticality under process variations.

Our simulation results demonstrate that:

- Previous techniques cannot guarantee the yields reported for uniform defects when defects are clustered. In contrast, the proposed technique can provide high yield in the realistic scenario of defect clustering.
- The proposed technique reduces delay overhead on the critical paths by 20.71% on average compared to a stress-unaware solution.
- D2D process variations in 3D ICs impact s-TSV allocation solutions significantly. The yield enhancement claimed in a variation-unaware method cannot be guaranteed for different process corner cases. Moreover, the variation-unaware approach could introduce excessive hardware cost for achieving a target manufacturing yield.

- Without considering the impact of temperature variation, thermal-unaware s-TSV allocation method would result in infeasible solution under a realistic temperature distribution, and the error in terms of delay overhead can be 6.41% on average.

The rest of this chapter is organized as follows. Preliminaries and related prior work are presented in Section 5.1. In Section 5.2, we present an example to highlight the motivation for this work. Section 5.3 describes the two-stage s-TSV allocation methodology in detail. In Section 5.4, we report simulation results. Finally, conclusions are drawn in Section 5.5.

5.1 Preliminaries and Related Work

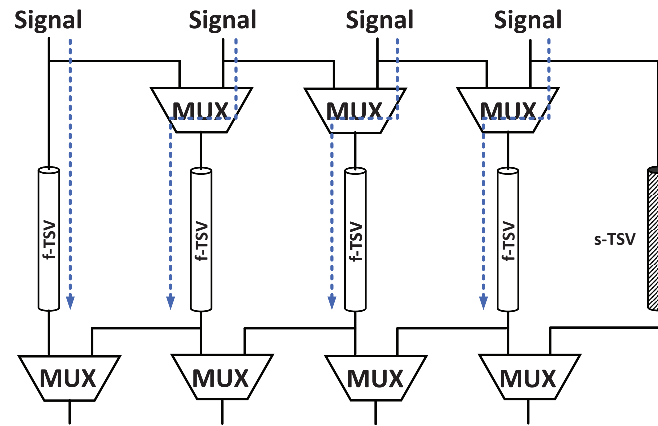
5.1.1 TSV fault tolerance

In this section, two techniques for TSV fault tolerance are briefly described; these methods are referred to as double TSV and shared s-TSV [168]. In the double TSV technique, each f-TSV is paired with an additional s-TSV to enhance yield and reliability. In the fault-free scenario, a signal is transferred through the two TSVs simultaneously. Once an f-TSV becomes faulty, there is still an s-TSV to pass the signal. Since the two TSVs are used to pass the same signal, no additional control circuits are required. However, due to the significant area overhead induced by s-TSVs, this technique is impractical.

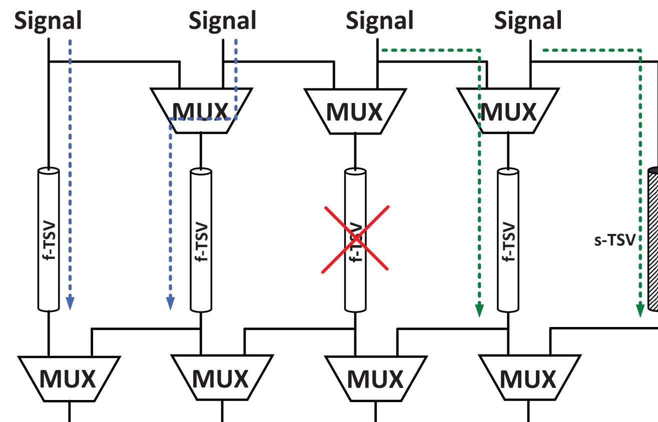
In order to reduce area overhead, the “shared s-TSV” technique has been proposed. In this technique, a set of f-TSVs is partitioned into several groups, and single or multiple s-TSV(s) are subsequently assigned to each group. By inserting MUXes and carefully designing the reconfigurable routing paths, the s-TSV(s) can be used to pass signal(s) in the presence of defective f-TSVs. In general, the shared s-TSV scheme can be classified into four categories [169]: i) shift-redundancy reconfiguration, ii) input and output reconfiguration, iii) re-route reconfiguration, and iv) only output reconfiguration. Figure 5.1 illustrates an example of the shift-redundancy reconfiguration. When all the f-TSVs are fault-free, the signals are transferred by them in Figure 5.1(a). Once an f-TSV fails, the signal corresponding to the faulty one has to be shifted, which causes all signals between the faulty TSV and the s-TSV to be shifted, as shown in Figure 5.1(b). Although this method reduces the area overhead compared to the double TSV technique, additional delay is introduced due to signal re-routing and the extra components.

5.1.2 Spare TSV allocation in different TSV placements

In 3D IC design, there are two different TSV placement styles, namely, uniform placement and non-uniform placement [165]. In uniform TSV placement, f-TSVs are placed uniformly, and they are considered as placement obstacles when logic cells (or IP blocks) are placed in the 3D placement stage. However, in non-uniform TSV placement, the f-TSVs are added to the 3D netlist as TSV cells, and then placed with the logic cells (or IP blocks) simultaneously during 3D placement. Compared to uniform placement,



(a) Operation when all f-TSVs are fault-free.



(b) Operation when the third f-TSV is faulty.

Figure 5.1: Illustration of the concept shared s-TSV technique.

non-uniform placement design can reduce wirelength by up to 17% [165], which leads to considerable latency decrease and power reduction.

For the above two placement styles, the allocation of s-TSVs is realized at different stages of the design flow. For the uniform case, since the locations of f-TSVs are determined before logic cells are placed, s-TSVs and the supporting infrastructure (i.e., MUXes and wires) can be inserted right after f-TSV planning but prior to the placement of the logic cells and detailed routing [160]. Therefore, it is not necessary to consider the whitespace constraint during s-TSV allocation. However, for non-uniform placement, the locations of f-TSVs and logic cells need to be determined simultaneously. Since allocating s-TSVs during the placement stage can result in significant perturbation to the routing of signal nets [162], the s-TSVs have to be inserted after the placement stage, and the whitespace constraint must be taken into account. Otherwise, the generated solution based on s-TSV allocation cannot guarantee repairability as explained in Section 5.2.

5.1.3 TSV-induced stress and stress-induced delay overhead

As mentioned in Section 5.1.1, although the shared s-TSV technique can reduce the area overhead introduced by s-TSVs, timing issues should be taken into account to guarantee the timing correctness of the circuit after TSV repair. When a signal is shifted, additional delay is introduced due to longer wires and extra components.

However, TSV-induced stress can also cause extra delay due to the significant carrier mobility variations in the vicinity of a TSV [93, 170]. Because of the mismatch in the coefficient of thermal expansion for copper and silicon, the fabrication of TSVs causes thermal-mechanical stress in the surrounding silicon. This stress can cause hole- and electron-mobility variations in the nearby transistors, which in turn results in significant delay change in the critical paths (up to 8.9% [100]). Therefore, it is necessary to consider the stress-induced delay overhead during s-TSV allocation.

For each transistor, its mobility change due to TSV-induced thermal stress can be calculated as [93]:

$$\frac{\Delta\mu}{\mu_0} = \frac{\Pi B \Delta\alpha \Delta T}{2} \sum_{s_j \in \mathbf{S}} \left(\frac{R_j}{r_j}\right)^2 \alpha\left(\theta_j\right) \tag{5.1}$$

where Π is a coefficient indicating the sensitivity of mobility change to applied stress, B is biaxial modulus, $\Delta\alpha$ is the CTE difference between copper and silicon, ΔT is the temperature difference between copper annealing and operating temperature. $\mathbf{S}$ represents the set of inserted s-TSVs that influence the transistor, R_j is the radius of s-TSV s_j, r_j is the distance from the center of s-TSV s_j to the transistor, and θ_j is defined as the angle between the center of the s-TSV s_j and the center of the channel of transistor when it is placed vertically. The variation in mobility can be translated into the variation in gate delay, which is calculated using the following sensitivity model [100]:

$$\Delta D_{\text{stress}} = \left(\frac{\partial D}{\partial \mu}\right) \Delta\mu\left(T\right) \tag{5.2}$$

where ΔD_{stress} is the gate delay change introduced by mobility variations, and $\partial D/\partial\mu$ is the sensitivity function of the delay to mobility variations. As shown in Equation (5.1) and Equation (5.2), temperature variation can impact the gate delay significantly, and this change can be caused by TSV-induced stress.

5.1.4 TSV defect clustering

TSV defect-distribution models can be viewed to be of two types, namely uniform defect distribution and clustered defect distribution. For the uniform TSV defect-distribution model, the defect probability of a specific TSV is assumed to be independent from that of the other TSVs. This model is valid for certain random defects such as void formation [171] and lamination due to thermal-induced stress [172]. However, many types of TSV defects appear during the imperfect bonding process. In addition, the winding level of the thinned wafer, the surface roughness, and cleanliness of silicon dies also impact the bonding quality of TSVs [173]. Consequently, if one TSV is faulty during the bonding process, it is more likely that its neighbouring TSVs are also defective; this is referred to as the "defect clustering effect" [158, 160]. This effect is not only pronounced in traditional semiconductor manufacturing, and it has also been reported for 3D ICs [166, 167]. The clustered defect-distribution model captures this more realistic scenario, where the presence of a defective TSV increases the probability of more defects in its close vicinity.

5.1.5 Related prior work

A number of s-TSV allocation strategies using shared s-TSVs have been proposed for trading-off between chip yield and hardware cost. The early work in [158] attempts to determine an optimal grouping ratio (i.e., the ratio of the number of s-TSVs to the number of f-TSVs) in each TSV group to minimize hardware cost while achieving high chip yield. However, this technique is only applicable to designs with uniform TSV placement, and neglects the delay overhead introduced by the fault tolerance solution.

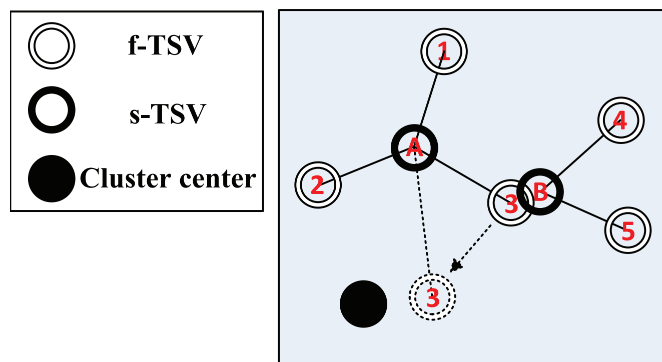
In [173], a TSV repair algorithm was proposed for yield enhancement under defect clustering. Compared to prior techniques that use a neighbouring s-TSV to repair a faulty f-TSV, the proposed technique allows a faulty f-TSV to be repaired by a distant s-TSV. Moreover, the timing impact of TSV repair was also taken into account by modeling the re-routing delay overhead. However, since the insertion of s-TSVs was assumed to be planned prior to the placement of logic cells, this technique is incompatible with non-uniform TSV placement.

Another technique dedicated to design with non-uniform TSV placement was presented in [161]. This technique attempted to minimize the delay overhead due to signal re-routing by optimally allocating s-TSVs to f-TSVs. However, the stress-induced additional delay was ignored, hence the delay overhead during s-TSV allocation was underestimated. Moreover, the chip yield, which is a primary motivation for s-TSV allocation, was not considered explicitly. Therefore, it is not clear if the generated solution can satisfy the target yield requirement without repeated trial-and-error.

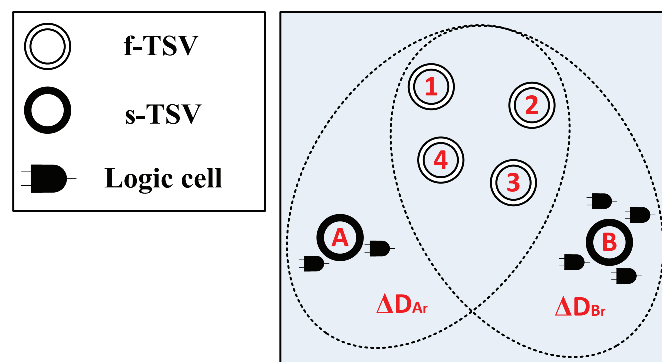
In [174], an optimal s-TSV allocation technique was proposed for non-uniform TSV placement. In this work, the area overhead introduced by the fault-tolerance solution was minimized under the constraints of yield and timing. Given the locations of all f-TSVs, the total number of s-TSVs is minimized by optimally allocating the s-TSVs to appropriate locations, forming redundancy structures with several f-TSVs. However, the stress-induced delay overhead was disregarded, hence it is difficult to guarantee the timing correctness of the circuit after TSV repair. Moreover, since defect clustering is ignored, the generated repair solution is likely to be ineffective under realistic defect distributions.

In conclusion, there are two major drawbacks of existing s-TSV allocation strategies for non-uniform TSV placement: i) the yield enhancement claimed in these methods cannot be guaranteed under realistic clustering defect distributions; ii) the stress-induced delay overhead has been ignored during s-TSV allocation, which makes it difficult to guarantee timing correctness after TSV repair.

5.2 Motivational Example



(a) The necessity of considering whitespace constraint.



(b) The necessity of considering stress-induced delay overhead.

Figure 5.2: Illustration for the motivational example.

The lack of effectiveness of previous techniques under a realistic defect distribution can be attributed to the fact that these techniques neglect whitespace constraints. In [162], it was assumed that s-TSVs can be placed anywhere on the chip. Since the s-TSVs are inserted after the placement stage, they cannot overlap with the placed f-TSVs. As shown in Figure 5.2(a), based on the appropriate location obtained from the s-TSV allocation, the s-TSV 'B' need to be inserted at this location, but this solution is inadmissible since it overlaps with the f-TSV '3'. To address this problem, [174] proposed to relocate these overlapped f-TSVs after s-TSV allocation by local legalization. As shown in Figure 5.2(a), f-TSV '3' is moved along the direction of the arrow to create space for the insertion of s-TSV 'B'.

However, due to the location-dependency, the defect probabilities of relocated f-TSVs will be changed. For example, as illustrated in Figure 5.2(a), the f-TSV '3' will be moved closer to a cluster center, which increases its defect probability significantly. Since the relocation is performed after s-TSV allocation, the generated repair solution from the s-TSV allocation cannot provide defect tolerance for f-TSVs with changed defect probabilities, which results in an unpredictable repair scenario. Therefore, it is necessary to consider the whitespace constraint during s-TSV allocation, i.e., s-TSVs can only be inserted in the whitespace.

Ignoring stress-induced delay overhead also results in an ineffective repair solution. In previous work (e.g., [161] and [174]), only the re-routing delay overhead was considered to guarantee timing correctness after TSV repair. However, the insertion of s-TSVs can also cause significant delay overhead due to TSV-induced stress, which should be carefully considered during s-TSV allocation.

Figure 5.2(b) shows that if we employ the shared s-TSV technique, the f-TSVs '1', '2', '3' and '4' are grouped together and assigned to the same s-TSV. For the s-TSV, there are two possible location candidates 'A' and 'B'. Suppose that ΔD_{A_r} and ΔD_{B_r} are the re-routing delay overheads when we place the s-TSV at these two location candidates, respectively. Assuming $\Delta D_{A_r} > \Delta D_{B_r}$, the s-TSV will be inserted at 'B' by stress-unaware allocation since this location leads to less re-routing delay overhead when we replace the faulty f-TSV. However, since there are more logic cells in the vicinity of 'B', inserting the s-TSV at 'B' will generate more stress and hence potentially affect the circuit delay. Therefore, to identify an optimal location candidate during s-TSV allocation, we must consider the stress-induced delay overhead.

5.3 Spare TSV Allocation Considering Defect Clustering

5.3.1 Problem statement

In this chapter, we solve the s-TSV allocation problem under defect clustering and using the shared s-TSV technique. The formal problem statement is as follows:

- **Input**: i) A 3D IC design where all f-TSVs have been placed; ii) target yield of the chip; iii) defect clustering parameters.

- **Constraints**: i) The chip yield should satisfy the target yield with the generated s-TSV allocation solution; ii) the s-TSVs should only be inserted in the whitespace between the placed cells; iii) the delay overhead of each f-TSV should be less than the upper bound of allowable delay overhead $\Delta D_{\max}$, which is a user-defined parameter.
- **Output**: an optimal s-TSV allocation solution, including: i) which f-TSVs are assigned to the same s-TSV(s); ii) the exact locations of the assigned s-TSVs.
- **Objective**: Minimize the hardware cost and the delay overhead induced by the fault-tolerance solution.

We propose a two-step optimization methodology to solve this problem. Before presenting the technique in detail, we introduce our notation and clarify assumptions related to the metrics.

5.3.2 Notation

We use the following symbols to facilitate the discussion.

- $\mathbf{F}$: a set of f-TSVs, where $|\mathbf{F}| = N_F$;
- f_i: an f-TSV with defect probability p_{f_i} and location (x_{f_i}, y_{f_i});
- $\mathbf{S}$: a set of s-TSVs candidates, where $|\mathbf{S}| = N_S$;
- s_j: an s-TSV candidate with defect probability p_{s_j} and location $\left(x_{s_j}, y_{s_j}\right)$;
- $\mathbf{G}_f$: a set of f-TSV groups, where $|\mathbf{G}_f| = N_{G_f}$;
- $\mathbf{G}_s$: a set of assigned s-TSV candidate groups ($\mathbf{G}_s \subseteq \mathbf{S}$), where $|\mathbf{G}_s| = N_{G_s}$;
- $\mathbf{G}$: the union of $\mathbf{G}_f$ and $\mathbf{G}_s$ ($\mathbf{G} = \mathbf{G}_f \cup \mathbf{G}_s$);
- g_k: a TSV group with yield Y_{g_k} ($g_k \in \mathbf{G}$), including an f-TSV sub-group $\mathbf{F}_{g_k}$ with N_{gf_k} f-TSV(s) and an s-TSV candidate sub-group $\mathbf{S}_{g_k}$ with N_{gs_k} s-TSV(s).

5.3.3 Assumptions related to metrics

Chip yield and TSV yield

The yield of a 3D IC $Y_{\text{3D-IC}}$ can be derived based on the cumulative yield property, as follows [34]:

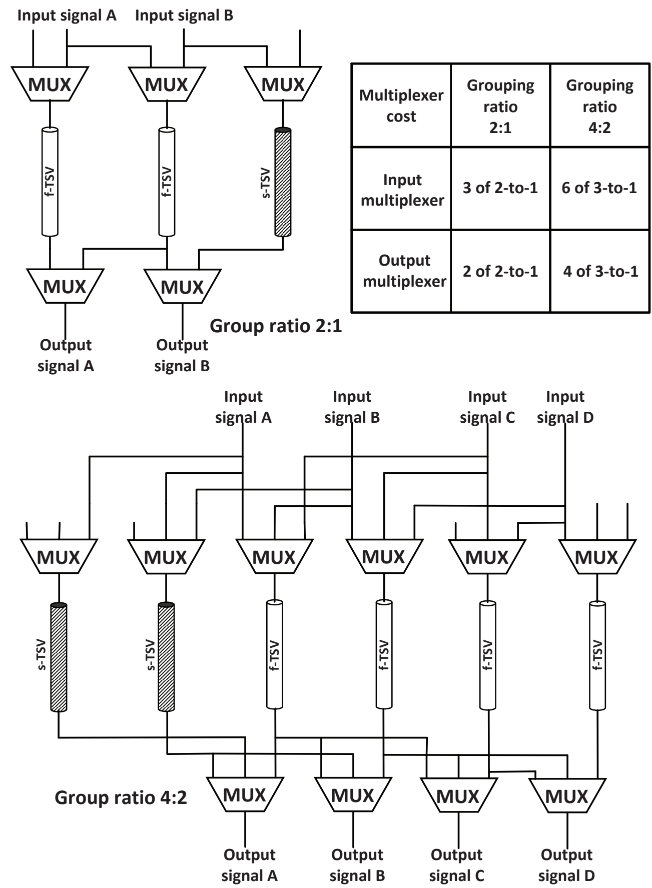
$$Y_{\text{3D-IC}} = Y_{\text{stack}} \prod_{i=1}^{N-1} Y_{\text{bonding}(i)} \prod_{i=1}^{N-1} Y_{\text{TSV}(i)} \tag{5.3}$$

where N is the number of device layers in the 3D chip, Y_{stack} is the overall yield of the N dies to be stacked, $Y_{\text{bonding}(i)}$ denotes the yield of the ith bonding step, and $Y_{\text{TSV}(i)}$

denotes the yield of the TSVs in the ith layer. In this chapter, we focus on improving the yield of 3D ICs in terms of overall TSV yield $Y_{\mathrm{TSV}(i)}$, as the other two parts in Equation (5.3) can be improved by either a better fabrication process or existing 2D defect-tolerance techniques [175].

Area overhead introduced due to fault-tolerance solution

In this chapter, we consider the hardware cost in terms of area overhead. The area overhead introduced by the fault-tolerance solution can be divided into several parts, including the area overhead due to inserted s-TSVs, related control logic (i.e., MUXes), and re-routing interconnect. The overhead is dominated by the first two parts [160]. In this chapter, we can estimate the total area overhead in term of the number of partitioned f-TSV groups N_{G_f}. The reasoning for this estimate is as follows.



Multiplexer cost	Grouping ratio 2:1	Grouping ratio 4:2
Input multiplexer	3 of 2-to-1	6 of 3-to-1
Output multiplexer	2 of 2-to-1	4 of 3-to-1

Figure 5.3: Illustration of the MUX configuration for two grouping ratios.

In Figure 5.3, the MUX configurations for two different grouping ratios (2:1 and 4:2) are shown. Here, the grouping ratio is given by $GR_k = N_{gf_k}{:}N_{gs_k}$. In this work, we

assume that a shift-redundancy reconfiguration scheme [169] is adopted in each group for signal re-routing. However, the proposed approach is generic, and it is not restricted to specified reconfiguration scheme. From Figure 5.3, the number of required MUXes N_{MUX_k} in group g_k is:

$$\begin{aligned} N_{\mathrm{MUX}_k} &= (N_{gf_k} + N_{gs_k} + N_{out_k}) \times \mathrm{MUX}_{(N_{gs_k}+1)-\mathrm{to}-1} \\ &= (2N_{gf_k} + N_{gs_k}) \times \mathrm{MUX}_{(N_{gs_k}+1)-\mathrm{to}-1} \end{aligned} \tag{5.4}$$

Here, N_{out_k} is the number of output signals, which is equal to N_{gf_k}. Note that the type of the MUX is related to N_{gs_k}.

Assume that a total of N_F f-TSVs are partitioned into N_{G_f} groups. Then the total number of MUXes required in this 3D IC (N_{MUX}) can be calculated as:

$$N_{\mathrm{MUX}} = \sum_{k=1}^{N_{G_f}} N_{\mathrm{MUX}_k} = \sum_{k=1}^{N_{G_f}} (2N_{gf_k} + N_{gs}) \tag{5.5}$$

Note that we insert the same number of s-TSVs N_{gs} in each group. Since $\sum_{k=1}^{N_{G_f}} N_{gf_k} = N_F$, N_{MUX} can be expressed as $2N_F + N_{G_f}N_{gs}$. Therefore, the total area overhead A_{total} is:

$$\begin{aligned} A_{\mathrm{total}} &= A_{\mathrm{spare}}N_{\mathrm{spare}} + A_{\mathrm{MUX}_{(N_{gs}+1)-\mathrm{to}-1}}N_{\mathrm{MUX}} \\ &= \left(A_{\mathrm{MUX}_{(N_{gs}+1)-\mathrm{to}-1}} + A_{\mathrm{spare}}\right) N_{G_f}N_{gs} \\ &\qquad + 2A_{\mathrm{MUX}_{(N_{gs}+1)-\mathrm{to}-1}}N_F \end{aligned} \tag{5.6}$$

Based on Equation (6.5), we conclude that for a fixed N_{gs}, minimizing the total area overhead A_{total} is equivalent to minimizing the number of the partitioned f-TSV groups N_{G_f}.

5.3.4 Overview of the proposed methodology

The overall flow of our approach is illustrated in Figure 5.4. As shown in this figure, the proposed method can be divided into three parts: 1) fault-map generation; 2) global partitioning; 3) detailed insertion of s-TSVs.

During fault-map generation, the defect probability of each f-TSV and s-TSV candidate is calculated using a clustered defect-distribution model. After obtaining the locations of f-TSVs and s-TSV candidates using the placement tool and the whitespace extractor, respectively, the defect probability for both types of TSVs can be determined.

Based on the generated defect probabilities, the f-TSVs are then partitioned into several groups under the constraint of target yield. As discussed in Section 6.4.3, the objective of global partitioning is to minimize the number of f-TSV groups. We use a greedy algorithm to solve this problem. Following this step, s-TSV(s) are inserted in the available location candidate(s) for each f-TSV group with the same number to

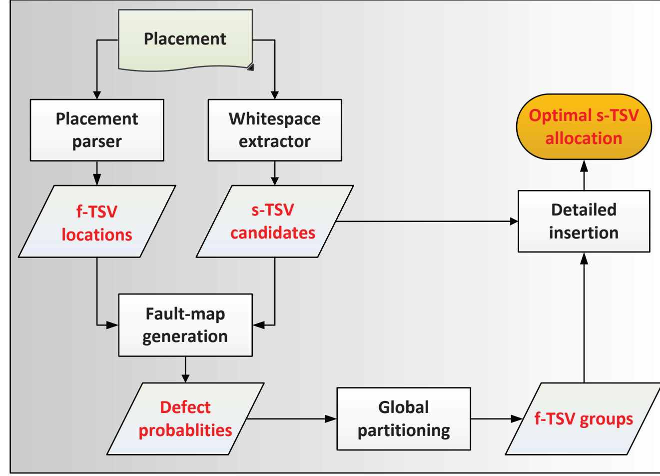


Figure 5.4: Overall flow of the proposed methodology.

minimize the delay overhead induced by fault-tolerant mechanism. In this step, an ILP formulation is used to generate an optimal s-TSV allocation. Next, we discuss the three steps in more detail.

5.3.5 Fault-map generation

The objective of this stage is to determine the defect probability for each f-TSV as well as each s-TSV candidate. In reality, defective TSVs tend to cluster together rather than being uniformly distributed. In [176], this defect-clustering effect is modeled; the defect probability is inversely proportional to the distance from existing defects, which are regarded as cluster centers [177]. If there are already N_c cluster centers, the defect probability of TSV_i, P_i, can be expressed as [158]:

$$P_i = p \cdot \left(1 + \sum_{j=1}^{N_c} \left(\frac{1}{d_{ij}}\right)^{\alpha}\right) \tag{5.7}$$

where p is the single TSV failure rate [158], and d_{ij} is the distance between TSV_i and the jth cluster center. α is the clustering coefficient, and a larger value of α implies more clustering. As shown by Equation (5.7), the TSV location information should be taken into account under the clustered-defect distribution.

For f-TSVs, the location information can be obtained from the placement of each tier in the 3D IC. For s-TSVs, as discussed in Section 5.2, they can only be inserted in whitespaces. Therefore, a branch-and-bound based approach is used to extract the whitespace [178], and then the location candidates for the s-TSVs can be determined based on the amount of whitespace and the area of a single TSV.

To model the clustering effect, the compound Poisson distribution is widely used [179]. In this model, the defect count follows Poisson distribution compounded with a Gamma

function presenting the distribution of defect density. Since f-TSVs and s-TSVs are assumed to be placed irregularly in this work, we need to construct a "virtual" regular TSV grid in order to find the defect-cluster centers among all f-TSVs and s-TSV insertion candidates. Here we use a similar technique proposed in [160]: the basic idea is to map all f-TSVs and s-TSV candidates into a regular TSV grid while maintaining the relative position of TSVs (The details can be found in [160]). Afterward, the cluster centers can be identified based on this constructed regular grid. As in [177], we assume that: i) the number of cluster centers is a random variable following Poission distribution and ii) they are uniformly distributed across the TSV grid. After determining the cluster centers among the f-TSVs and s-TSV candidates, the defect probabilities for the other TSVs can be calculated using Equation (5.7). Note that we do not insert the s-TSVs at the locations that are identified as cluster centers due to the high defect probabilities associated with them. In this way, the defect probability of each f-TSV and inserted s-TSV can be estimated pessimistically, which can however lead to excessive hardware overhead in terms of the inserted s-TSVs. Meanwhile, the target reliability requirements will not be jeopardized. However, such inaccuracies are inevitable without a trial-and-error approach. Although we can obtain a more accurate fault-map by inserting s-TSVs into all candidates identified as cluster centers, this can result in a more costly repair solution as these s-TSVs cannot be used as redundancy for yield enhancement.

5.3.6 Global partitioning

In this step, f-TSVs are partitioned into several groups, which can be of different sizes. In order to minimize area overhead under the constraint of target yield. This problem can be formulated as follows:

- **Input**: i) a set of f-TSVs $\mathbf{F}$; ii) a set of s-TSV candidates $\mathbf{S}$; iii) the target yield Y_{target}; iv) the upper bound of allowable delay overhead ΔD_{max}.
- **Constraint**: i) $Y_{\text{TSV}} \geq Y_{\text{target}}$; ii) the delay overhead of each f-TSV should be not more than ΔD_{max}.
- **Output**: A set $\mathbf{G}_f$ of f-TSV groups that partitions $\mathbf{F}$.
- **Objective**: Minimize N_{G_f}.

The major challenge inherent in this problem is to undertake a trade-off analysis between the yield improvement for a 3D IC and the corresponding area overhead. By placing fewer f-TSVs in a group, higher yield can be guaranteed due to the fixed N_{gs} of each group, but this solution will also result in higher area overhead. In this section, we first discuss the computational complexity of this global partitioning problem, and provide a proof of $\mathcal{NP}$-hardness. As this global partitioning problem can be proven to be $\mathcal{NP}$-hard, a greedy heuristic algorithm is proposed to solve this problem through a careful selection of the number of f-TSVs in each group. The basic idea is to merge the two f-TSV groups with the highest yield together to reduce N_{G_f} until Y_{TSV} is reduced to Y_{target}.

Calculation of yield

Before presenting the proof of $\mathcal{NP}$-hardness and heuristic algorithm in detail, it is necessary to consider how to calculate Y_{TSV} under clustered defects. By multiplying all the TSV group yield Y_{g_k}, Y_{TSV} can be calculated as:

$$Y_{\mathrm{TSV}} = \prod_{k=1}^{N_{G_f}} Y_{g_k} \tag{5.8}$$

while Y_{g_k} can be obtained as:

$$Y_{g_k} = 1 - p_{g_{k_{\text{non_repair}}}} \tag{5.9}$$

where $p_{g_{k_{\text{non_repair}}}}$ is the probability that the group g_k cannot be repaired. Note that here we ignore the yield loss due to the additional control logic (i.e., MUX) when evaluating Y_{g_k}. This assumption is justified because the number of added transistors in the control logic is negligible compared to the entire design. To calculate $p_{g_{k_{\text{non_repair}}}}$, the basic idea is to search for all the possible scenarios in which the group cannot be required, and accumulate the probabilities for these scenarios as $p_{k_{\text{non_repair}}}$. Assuming that there are N TSVs (including K f-TSV(s) and $N-K$ s-TSV(s)) in g_k, this group cannot be repaired if and only if the number of non-defective TSV(s) is less than K, in which the case of a single defective TSV is also included. Therefore, by exhaustively searching all these possible cases that a group cannot be repaired and summing up the probability of each case, $p_{g_{k_{\text{non_repair}}}}$ can be calculated, and then the yield for this group can be computed according to Equation (5.9).

Here each TSV group can also be treated as a "K-out-of-N:Good" system with components having non-identical defect probabilities [180], in which an N-component system that works if and only if at least K of the N components are "good". In order to calculate the group yield, an algorithm has been proposed in [181], whose computational complexity is $\mathcal{O}(N^2)$ [180].

Proof of $\mathcal{NP}$-hardness

In the following, a formal proof is provided to show the $\mathcal{NP}$-hardness of the global partitioning problem. The original global f-TSV partition problem is to partition the set of given f-TSVs into a minimum number of groups under the constraints of target yield and re-routing delay. In order to prove that this optimization problem is $\mathcal{NP}$-hard, we must prove its corresponding decision problem is $\mathcal{NP}$-complete, which can be defined as follows.

Definition 1: Yield-constrained and delay-constrained minimum TSV partition problem (**YDMTPP**)

- INSTANCE: A set of f-TSVs $\mathbf{F} = \{f_i\}$ in which each f-TSV has defect probability p_{f_i}, a set of s-TSV candidates $\mathbf{S} = \{s_j\}$ in which each candidate has defect

probability p_{s_j}, and a positive integer $M \leq |\mathbf{F}|$, where i) the set of TSV groups $\mathbf{G} = \{g_k\}$ is a collection of several disjoint subsets of $\mathbf{F}$ and $\mathbf{S}$ (i.e., $\mathbf{G}_f$ and $\mathbf{G}_s$), in which the size of each s-TSV candidate subset is not less than N_{gs}; ii) the delay overhead of each f-TSV f_i during signal re-routing should not more than $\Delta D_{\max}$ [1]; iii) the achieved yield Y_{achieved} of the entire system should not be less than Y_{target};

- QUESTION: Is there a set of all f-TSV subsets in $\mathbf{G}$ (i.e., $\mathbf{G}_f \subset \mathbf{G}$) partitioning $\mathbf{F}$ with $|\mathbf{G}_f| \leq M$?

Before we discuss the general problem, let us consider the following restricted variant of YDMTPP by allowing only the instances where: i) $p_{f_i} = p_{s_j} = 0$ for all i, j, and ii) in each TSV group g_k, the distance between any two f-TSVs is not more than the distance between any f-TSV and the assigned s-TSV candidate:

Definition 2: Delay-constrained minimum TSV partition problem (**DMTPP**)

- INSTANCE: A set of f-TSVs $\mathbf{F} = \{f_i\}$, a set of s-TSV candidates $\mathbf{S} = \{s_j\}$, and a positive integer $M \leq |\mathbf{F}|$, where i) the set of TSV groups $\mathbf{G} = \{g_k\}$ is a collection of several disjoint subsets of $\mathbf{F}$ and $\mathbf{S}$ (i.e., $\mathbf{G}_f$ and $\mathbf{G}_s$), in which the size of each s-TSV candidate subset is not less than N_{gs}; ii) in each TSV group g_k, the distance between any two f-TSVs is not more than the distance between any f-TSV and s-TSV candidate; iii) the delay overhead of each f-TSV f_i during signal re-routing should not more than $\Delta D_{\max}$;
- QUESTION: Is there a set of all f-TSV subsets in $\mathbf{G}$ (i.e., $\mathbf{G}_f \subset \mathbf{G}$) partitioning $\mathbf{F}$ with $|\mathbf{G}_f| \leq M$?

In order to prove YDMTPP is $\mathcal{NP}$-complete, we just need to prove that DMTPP is $\mathcal{NP}$-complete.

Theorem 1: The DMTPP is $\mathcal{NP}$-complete.

Proof 1: To show DMTPP is in $\mathcal{NP}$, our verifier takes a set of f-TSVs $\mathbf{F}$, a set of s-TSV candidates $\mathbf{S}$, and a set of TSV group $\mathbf{G}_f$, and can check in polynomial time whether $\mathbf{G}_f$ is a proper partition of a certain size by checking: i) if all f-TSV subsets in $\mathbf{G}_f$ are disjoint, ii) if all f-TSV subsets in $\mathbf{G}_f$ cover $\mathbf{F}$, iii) if the size of the s-TSV candidate subset for each TSV group is not less than N_{gs}, iv) if all s-TSV candidate subsets are disjoint, and v) if the delay overhead of each f-TSV f_i during signal re-routing is not more than $\Delta D_{\max}$. In order to show this problem is $\mathcal{NP}$-hard, we provide a polynomial-time reduction from the graph-coloring problem to this DMTPP.

Definition 3: M-Coloring problem (**MCOL**)

- INSTANCE: An undirected graph $G = (V(G), E(G))$, and a positive integer $M \leq |V(G)|$;

[1] Please note that here we only consider the delay overhead introduced by additional wirelength during signal re-routing between f-TSV and s-TSV since the MUX-induced delay overhead is only determined by the number of assigned s-TSV(s) in each group N_{gs}.

- QUESTION: Is G M-colorable, that is, does there exist a function $f : V(G) \rightarrow \{1, 2, ..., M\}$ such that $f(u) \neq f(v)$ whenever $\{u, v\} \in E(G)$?

Let the undirected graph $G = (V(G), E(G))$ and the positive integer $M \leq |V(G)|$ constitute an arbitrary instance Φ of MCOL. And we construct a reduction τ to an instance $\tau(\Phi)$ of the DMTPP as follows. First, we add MN_{gs} vertex(es) $\{s_j\}$ into G, such that all the MN_{gs} vertex(es) is (are) sufficiently near to all vertices in $V(G)$. Next, for every N_{gs} vertex(es) in $\{s_j\}$, we connect it (them) to other $(M-1)N_{gs}$ vertex(es) in $\{s_j\}$ (if any). In other words, there is (are) M independent set(s) in $\{s_j\}$ after connection. This modified graph is called $G' = \left(V\left(G'\right), E\left(G'\right)\right)$. Accordingly, the instance of DMTPP $\tau(\Phi)$ can be specified by:

- $\mathbf{F} := V\left(\overline{G}\right)$
- $\mathbf{S} := \{s_j\}$ $(j = 1, 2, \ldots, MN_{gs})$

Claim 1: There is a feasible solution to an instance Φ of the MCOL problem if and only if the instance $\tau(\Phi)$ of the DMTPP has its solution with value M.

To prove the condition is necessary ("only if"). In fact, for any feasible solution of an instance Φ of the MCOL problem (i.e., G is M-colorable), $V(G)$ can be partitioned into M independent set(s). Since there is (are) also M independent set(s) in $\{s_j\}$, G' can also be partitioned into M independent set(s) (i.e., G' is also M-colorable). Due to the complementary relationship between independent set and clique, any independent set can become a clique by taking the complement graph. Therefore, if G' can be partitioned into M independent set(s), then $\overline{G'}$ can be partitioned into M cliques $C = \{C_i\}$ $(i = 1, 2, \ldots, M)$. Here each clique can be treated as an f-TSV group with N_{gs} s-TSV candidate(s). Since we assumed that all the MN_{gs} vertex(es) in $\{s_j\}$ is (are) sufficiently near to all vertices in $V(G)$, the constraints of delay overhead in DMTPP are always satisfied for each group. Consequently, there is a set of f-TSV groups that can partition all f-TSVs into M subsets.
To prove the condition is sufficient ("if"). If there is an f-TSV group set that can partition all f-TSVs into M subsets, $\overline{G}$ can be partitioned by M cliques. Therefore, G has a partition into M independent sets, and hence is M-colorable.

Hence, from Claim 1 and the fact that the reduction takes polynomial time, Theorem 1 holds for $M \geq 3$.

Theorem 2: The YDMTPP is $\mathcal{NP}$-complete.

Proof 2: Restrict to DMTPP: Make that i) $p_{f_i} = p_{s_j} = 0$ for all i, j, and ii) the distance between any two f-TSVs is not more than the distance between any f-TSV and the assigned s-TSV candidate in each TSV group.

As we have proved that the YDMTPP is $\mathcal{NP}$-complete, its corresponding optimization problem (i.e., the global partitioning problem) is $\mathcal{NP}$-hard.

Our heuristic

Although the global partitioning problem is $\mathcal{NP}$-hard, an efficient heuristic algorithm can be developed. The objective during this stage is to minimize the number of partitioned groups N_{G_f} under the constraints of target yield. To this end, a greedy algorithm is used (as shown in Algorithm 2). The first step is to perform an initial group partitioning, in which each f-TSV is treated as an individual group consisting of N_{gs} s-TSV(s). This way, the maximum number of s-TSVs required is $N_F \times N_{gs}$. If the required number is less than the total number of s-TSV candidates N_S, multiple f-TSVs with the lowest defect probabilities can be partitioned into the same group to reduce the number of the required s-TSVs.

After the initial partitioning, the yield of each group should be calculated; this step requires the information about the s-TSV(s) assigned to each group, including the corresponding number and defect probabilities. In this work, the same number of s-TSV(s), N_{gs}, is assigned for each group. Since the exact location of each assigned s-TSV will be determined in the next step, the defect probability cannot be obtained during the global partitioning step due to its location-dependency. Therefore, all the assigned s-TSVs are assumed to have the maximum defect probability among all the s-TSV candidates. This way, the yield for each group can be calculated. The reduction of N_{G_f} is realized iteratively until the existing TSV yield is reduced to the target one; we merge the two "available" groups with the highest yields together in each step to reduce N_{G_f} by 1.

The term "the two available groups" means that the generated group g_k from group merging should satisfy the condition:

$$\left|\mathbf{S}_{\mathbf{F}_{g_k}}\right| = \left|\bigcap_{f_i \in \mathbf{F}_{g_k}} \mathbf{S}_{f_i}\right| \geq N_{gs} \tag{5.10}$$

where $\mathbf{S}_{f_i}$ is the set of available candidates for f-TSV f_i and $\mathbf{S}_{\mathbf{F}_{g_k}}$ is the set of available candidates for f-TSV group $\mathbf{F}_{g_k}$. Equation (5.10) ensure that there are enough options for the succeeding s-TSV insertion step for each group under Constraint (iii) mentioned in Section 7.4.1. After global partitioning, the set of available candidates for each f-TSV f_i is updated as:

$$\text{if } f_i \in \mathbf{F}_{g_k} : \ \mathbf{S}_{f_i} = \mathbf{S}_{\mathbf{F}_{g_k}} \tag{5.11}$$

Here, an "available candidate" for each f-TSV f_i can be defined as follows: When rerouting the carried signal by f_i to s-TSV s_j, if the introduced delay overhead ΔD_{ij} is less than the upper bound of allowable delay overhead $\Delta D_{\max}$, s_j is identified as the so-called "available candidate" for f_i. ΔD_{ij} is originated from: i) the delay overhead introduced by control logic circuities (i.e., MUXes) ΔD_{MUX}, and ii) the additional interconnect delay due to signal re-routing ΔD_{DIST}. ΔD_{MUX} is determined by the number of s-TSVs in each group N_{gs}. As shown in Figure 5.3, the value of ΔD_{MUX} for each partitioned group can be given by:

$$\Delta D_{\text{MUX}} = 2 \times D_{\text{MUX}_{(N_{gs}+1)-\text{to}-1}} \tag{5.12}$$

In order to evaluate ΔD_{DIST}, we take the distance between f_i and s_j as the metric, and calculate ΔD_{DIST} by Elmore delay model, as in [161]. In addition, since the carried signal for each f-TSVs would be also re-routed through other f-TSVs in the adopted shift-redundancy reconfiguration scheme (as shown in Figure 5.1), it is necessary to guarantee that the delay overhead introduced by the signal re-routing between f-TSVs in each group is not more than the value of $\Delta D_{\max}$ (i.e., the upper bound of allowable delay overhead) during the group merging.

Note that here $\Delta D_{\max}$ should be set to different values for different f-TSVs according to their timing requirements. In other words, the constraints of timing slack should be taken into account during global partitioning in order to avoid the timing violation when re-routing a signal with distant TSV and achieve timing closure. In our problem formulation, the constraints of timing slack can be extracted through the following steps:

- Perform STA;
- Extract the timing budget *Timing Budget* (TB) for each f-TSV by delay budgeting technique (e.g., zero-slack algorithm proposed in [182]);
- Determine the upper bound of delay overhead for each f-TSV according to the extracted timing budget information.

With the consideration of timing slack constraints, all f-TSVs can be distinguished for two categories:

- "unmergeable" f-TSVs with insufficient timing budgets, i.e.,

$$\mathrm{TB}\left(f_i\right) < 2D_{\mathrm{MUX}_{(N_{gs}+1)-\mathrm{to}-1}} \tag{5.13}$$

- "mergeable" f-TSVs with sufficient timing budgets, i.e.,

$$\mathrm{TB}\left(f_i\right) \geq 2D_{\mathrm{MUX}_{(N_{gs}+1)-\mathrm{to}-1}} \tag{5.14}$$

After the identification of timing criticality for each f-TSV, different strategies are adopted for different types of f-TSVs during the group merging.

- For the "unmergeable" f-TSVs, we partition each of them into individual group during global partitioning, and don't allow them to be merged with other groups afterwards due to their limited timing budgets.
- The "mergeable" f-TSVs can be merged with other f-TSVs during global partitioning under the constraints of given timing budgets.

Therefore, for each f-TSV $f_i \in \mathbf{F}$, the value of $\Delta D_{\max}$ can be determined according to its timing criticality:

$$\Delta D_{\max}\left(f_i\right) = \begin{cases} \Delta D_{\mathrm{DIST}}^{\max}\left(f_i\right), \text{ if } f_i \text{ is unmergeable} \\ \Delta D_{\mathrm{MUX}} + \Delta D_{\mathrm{DIST}}^{\max}\left(f_i\right), \text{ otherwise} \end{cases} \tag{5.15}$$

Here $\Delta D_{\mathrm{DIST}}^{\max}(f_i)$ is the maximum interconnect delay when the signal carried by f_i is re-routed through the most distant available TSV in the design, and its value can be determined by the following equation for a given N_{gs} (i.e., a deterministic ΔD_{MUX}):

$$\forall f_i \in \mathbf{F}: \ \Delta D_{\max}(f_i) \leq \mathrm{TB}(f_i) \tag{5.16}$$

In particular, for the "mergeable" f-TSVs with relaxed timing budgets in an unbalanced design, we set an identical $\Delta D_{\max}$ for all of them under a threshold value according to the chip size, which is beneficial for signal re-routing between f-TSVs and the assigned s-TSVs. Note that the stress-induced delay overhead is not considered here as the appropriate candidates for s-TSV insertion will be determined in the following step.

Algorithm 1 Greedy algorithm for minimizing N_{G_f}

Input: $\mathbf{F}$, $\mathbf{S}$, Y_{target}, N_{gs}, and $\Delta D_{\max}$
Output: $\mathbf{G}_f$
1: perform initial partitioning on $\mathbf{F}$, obtain $\mathbf{G}_f^0$;
2: **repeat**
3: sort $\mathbf{G}_f^0$ based on group yield in descending order;
4: select the first two available groups in $\mathbf{G}_f^0$;
5: merge the selected groups together as g_k;
6: delete N_{gs} s-TSV(s) in g_k;
7: re-calculate the yield of g_k;
8: $\mathbf{G}_f^0 = \mathbf{G}_f^0 \cup \{g_k\}$;
9: delete the selected two groups in $\mathbf{G}_f^0$;
10: recalculate the yield of $\mathbf{G}_f^0$ as Y_{existing};
11: **until** $Y_{\mathrm{existing}} > Y_{\mathrm{target}}$
12: $\mathbf{G}_f = \mathbf{G}_f^0$;
13: output $\mathbf{G}_f$;

5.3.7 Detailed insertion of spare TSVs

After global partitioning, the f-TSVs are already partitioned into several groups with a minimum number of inserted s-TSVs in total under the constraint of a target yield. The objective of this insertion stage is to minimize the delay overhead incurred after TSV repair by inserting the s-TSVs in the appropriate locations. This problem can be formally stated as follows:

- **Input**: i) a set of f-TSV groups $\mathbf{G}_f = \{\mathbf{F}_{g_k}\}$, which partitions $\mathbf{F}$; ii) a set of s-TSV candidates $\mathbf{S}$; iii) N_{gs}.

- **Output**: an optimal s-TSV allocation solution, which indicates that: i) which s-TSV candidates are selected; ii) which f-TSVs are mapped to the selected candidates.

- **Objective**: the overall circuit delay overhead introduced by the s-TSV allocation solution is minimized, which includes: i) re-routing delay overhead; ii) stress-induced delay overhead.

In order to accurately evaluate the impact of delay overhead on circuit performance, we consider the delays of all critical and near-critical paths (i.e., VPs) during the insertion of s-TSVs. The reason is that the delay increase of a near-critical path could be more than that of a critical path with the consideration of stress-induced delay overhead. The list of VPs can be extracted by STA of the circuit with a threshold value either in path slack or in path number. We solve the above problem using the following ILP formulation.

Let x_{ij} be a binary variable defined as follows: $x_{ij} = 1$ if f-TSV f_i is mapped to s-TSV candidate s_j, and $x_{ij} = 0$ otherwise. Each f-TSV group $\mathbf{F}_{g_k} \in \mathbf{G}_f$ must satisfy the following constraint:

$$\forall f_i \in \mathbf{F}_{g_k} : \sum_{j=1}^{\left|\mathbf{S}_{\mathbf{F}_{g_k}}\right|} x_{ij} = N_{gs} \tag{5.17}$$

$$\sum_{j=1}^{\left|\mathbf{S}_{\mathbf{F}_{g_k}}\right|} \left(\prod_{i=1}^{\left|\mathbf{F}_{g_k}\right|} x_{ij} \right) = N_{gs} \tag{5.18}$$

Equation (7.4) implies that for each f-TSV $f_i \in \mathbf{F}_{g_k}$, the number of mapped s-TSV candidates should be N_{gs}, which is a user-defined parameter. Moreover, the candidate can only be selected from the available candidate set $\mathbf{S}_{\mathbf{F}_{g_k}}$ of group $\mathbf{F}_{g_k}$. Equation (7.5) guarantees that all the f-TSVs from the same group can be mapped to the same candidate(s). Moreover, the products of binary variables in Equation (7.4) can be eliminated in the linear programming formulation through standard techniques [111].

Next, we state the objective function (OF) of the ILP formulation. The objective at this stage is to optimize (i.e., minimize) the worst-case delay among all the VPs of the circuit with the consideration of the delay overhead introduced by s-TSV insertion. The post-insertion path delay of a VP can be presented as:

$$D\left(\mathrm{VP}_t\right) = D_0 + \sum_{\forall c_s \in \mathrm{VP}_t} \Delta D\left(c_s\right) + \sum_{\forall f_s \in \mathrm{VP}_t} \Delta D\left(f_s\right) \tag{5.19}$$

where VP_t is the tth VP in the set of extracted vulnerable paths $\mathbf{VP}$, and c_s and f_s are the logic cells and f-TSVs in VP_t, respectively. $\Delta D\left(c\right)$ is the stress-induced delay overhead on logic cell c introduced by s-TSV insertion, which can be expressed as:

$$\Delta D\left(c\right) = \sum_{j=1}^{|\mathbf{S}|} \Delta D_j^{\text{stress}}\left(c\right) \cdot \mathrm{OR}\left(x_{1j}, x_{2j}, \ldots, x_{|\mathbf{F}|j}\right) \tag{5.20}$$

here $\Delta D_j^{\text{stress}}\left(c\right)$ is the stress-induced delay overhead of logic cell c when inserting s-TSV at the j$^{\text{th}}$ candidate in $\mathbf{S}$, which can be calculated by Equations (5.2) and (5.1). OR

refers to logic functions $f = \text{OR}\left(\alpha_1, \alpha_2, \ldots, \alpha_w\right)$, which can be linearized as follows [183]:

$$\begin{aligned} &f \le \alpha_1 + \alpha_2 + \ldots + \alpha_w \\ &f \ge \left(\alpha_1 + \alpha_2 + \ldots + \alpha_w\right)/w \end{aligned} \tag{5.21}$$

Equation (5.20) implies that the stress-induced delay overhead of each logic cell can be calculated by accumulating the impact of each inserted s-TSV on it. In addition, the re-routing delay overhead $\Delta D\left(f\right)$ for each f-TSV $f_i \in \mathbf{F}$ can be expressed as:

$$\Delta D\left(f_i\right) = \max\left(\left\{D_{ij}^{\text{re-routing}} x_{ij} : s_j \in \mathbf{S}_{f_i}\right\}\right) \tag{5.22}$$

where $D_{ij}^{\text{re-routing}}$ is the additional interconnect delay when signal is re-routed between f_i and s_j, which can be calculated using Elmore delay model [161]. Since each f-TSV could be connected with multiple s-TSVs for effective yield enhancement, we set the worst-case re-routing delay overhead as optimization objective at this stage. Note that here only the interconnect optimization between f-TSVs and s-TSVs is considered since the constraints of delay overhead introduced by "*f-TSV*-to-*f-TSV*" signal re-routing have been taken into account implicitly during global partitioning. First, Equation (5.19) gives rise to the following constraints:

$$\forall \text{VP}_t \in \mathbf{VP} : \ D\left(\text{VP}_t\right) \le \text{slack}\left(\text{VP}_t\right) \tag{5.23}$$

here slack is the slack value of each VP, which is the difference between require time and arrive time. Equation (7.6) guarantees that all VPs still satisfy the given timing requirement with the consideration of re-routing and thermal stress-induced delay overheads. Second, the objective function OF can be given by:

$$\text{Minimize} \quad \text{OF} = \max_{t=0}^{|\mathbf{VP}|}\{D\left(\text{VP}_t\right)\} \tag{5.24}$$

The above minmax objective can be simply linearized using standard techniques associated with ILP modeling [161]. Therefore, combining Equations (7.4), (7.5), (7.6) and (5.24), we obtain an ILP model to minimize the delay overhead, and this model can be solved by an ILP solver.

5.4 Simulation Results

5.4.1 Simulation setup

We assume four-layer 3D integration scheme as test case in this work. The benchmark circuits were selected from the ITC'99 benchmark suite [114] and the experiments were conducted using the TSMC 40-nm library. The detailed information of each benchmark, including gate count, VP count, and the number of f-TSVs and s-TSV candidates, are listed in Table 5.1. The experiments were performed on a server with four AMD Opteron 6174 processors and 256GB RAM. The netlist and the layout (in 2D) for each design are

extracted using Synopsys Design Compiler and Cadence SoC Encounter, respectively. The UCLA 3D physical design flow "3D-Craft" was utilized to convert the designs to 3D ICs [118]. Based on the 3D placement, the locations of the f-TSVs and s-TSV candidates can be extracted, which serves as the input data for global partitioning and detailed s-TSV insertion. For both f-TSVs and s-TSVs, the TSV cell size is 2.47 μm $\times$ 2.47 μm. The ILP solver CPLEX [119] was used to obtain optimal s-TSV allocation solutions.

Table 5.1: Design Statistics for all benchmarks.

Benchmark	# Gates	# VPs	# f-TSVs	# s-TSV candidates
b17	22,003	47	1,563	2,278
b18	67,319	164	4,286	8,517
b19	117,506	111	5,477	11,516
b20	18,817	23	2,093	2,471
b21	20,798	31	2,279	2,481
b22	29,027	79	2,789	3,419

In order to accurately estimate the stress-induced delay overhead, the thermal profile of the entire design needs to be derived. After creating a top-level Verilog netlist for the design, post-synthesis simulations were conducted in Modelsim with a testbench containing 10^6 random input vectors. This way, the switching activity of each net can be extracted. A more accurate estimation can be achieved by sophisticated approaches such as the one proposed in [184]. Afterwards, the extracted *Switching Activity Interchange Format* (SAIF) file was forwarded to Power Compiler in order to obtain the power consumption of each logic cell. Based on this information and the corresponding layout file, thermal analysis can be carried out using the 3D *HotSpot* [149]. According to the derived thermal profile, the stress-induced delay variation can be calculated using Equations (5.2) and (5.1). In this work, we assume that the copper annealing temperature is 200 °C [117]. Due to the non-linearity of the sensitivity function in Equation (5.2), it is stored as a LUT rather than a constant value. During the calculation of delay change, linear interpolation is used between the stored points in the LUT.

In order to obtain TBs for each f-TSV, STA needs to be performed for each design. To this end, the SPEF file of each tier was extracted by performing parasitic extraction after routing. Combined with the extracted parasitics information, the top-level Verilog file consisting of the interconnections among tiers is fed into STA tool (e.g., Synopsys Primetime) to obtain the delay of each path. The target clock period CLK is set to:

$$\text{CLK} = (1 + c\%) \times \text{LPD} \tag{5.25}$$

where *Longest Path Delay* (LPD) is the longest path delay without considering s-TSV insertion, and $c\%\times$LPD is equal to the sum of TBs for the f-TSVs located on the critical paths. In this work, we take $c\%\times$LPD as a metric for timing budget of f-TSVs. For each path, we assume that its slack is divided uniformly for each TSV located on it. Among all the paths, the ones with maximum 9% difference on path delay value relative to the

critical path are extracted during STA and labeled as VPs. This assumption is consistent with the previous work suggesting near-critical paths are possible to become critical due to thermal stress introduced by TSV insertion [100]. In a realistic design, there could be lots of VPs since the paths are fully balanced. Consequently, it is infeasible to take all the VPs into account in the ILP formulation. A possible solution to this issue lies in the selection of only a small set of representative critical paths from the numerous VPs [109].

5.4.2 Impact of design knobs on spare TSV allocation

There are two user-defined parameters in the proposed technique, namely $\Delta D_{\max}$ and N_{gs}. In this section, we investigate their impact on the generated s-TSV allocation solution, and undertake the trade-off analysis among chip yield, hardware cost and delay overhead.

Impact of $\Delta D_{\max}$

$\Delta D_{\max}$ is a crucial parameter in the proposed technique. As discussed in Section 5.3.6, since the location candidates are determined *a priori* based on the availability of whitespaces, the varied $\Delta D_{\max}$ can change the size of $\mathbf{S}_{f_i}$ for each f-TSV f_i, which has a significant impact on the generated solution. Moreover, in order to guarantee timing closure, $\Delta D_{\max}$ should be determined under the constraints of timing requirement for each design. Here we present the impact of $\Delta D_{\max}$ on the hardware cost due to inserted s-TSVs.

The experiment was performed on benchmark b17. According to Equation (5.16), the $\Delta D_{\max}$ of each f-TSV is highly dependent on CLK (i.e., TB) and N_{gs} (i.e., ΔD_{MUX}). Figure 5.5 illustrates the number of "unmergeable" f-TSVs with varied N_{gs} and TBs. As shown, with the increased timing budget (i.e., $c\%$), more f-TSVs are recognized as "mergeable" from "unmergeable" category for all N_{gs} values, which provide more degree of freedom for s-TSV allocation. On the other hand, the number of "unmergeable" f-TSVs is increased with N_{gs} for a fixed timing budget due to the increased propagation delay of MUXes, which will be discussed in Section 5.4.2.

Next, we investigate the trade-off between timing performance and hardware cost in s-TSV allocation. The experiment was performed on benchmark b17 with $1,563$ f-TSVs to achieve the target yield $Y_{\text{TSV}} = 99.8\%$ under the clustered defect distribution, in which the clustering coefficient $\alpha = 2$ and the TSV defect probability p was set to 0.001. For each TSV group, N_{gs} is set to 1. Table 5.2 illustrates the relationship between the achieved yield and the total number of inserted s-TSVs. As shown in the table, with a strict timing requirement (i.e., small $c\%$), the number of the elements in $\mathbf{S}_{f_i}$ for each f_i is small. Therefore, it is difficult to satisfy Equation (5.10) during global partitioning, which results in infeasible solution (e.g., $c = 0\%, 5\%, 15\%$). As the requirement is relaxed, more degree of freedom is available for f-TSV partitioning, which yields an optimized solution in terms of the total number of inserted s-TSVs. However, the timing performance of the design is degraded. Accordingly, there is a trade-off

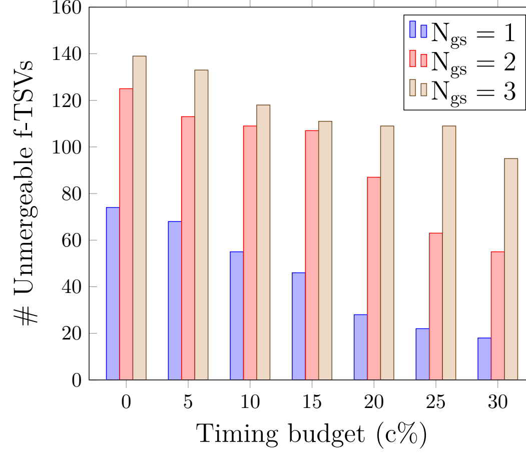


Figure 5.5: The number of unmergeable f-TSVs with varied N_{gs} and $c\%$ for benchmark b17.

between timing performance and hardware cost in s-TSV allocation. Consequently, the target clock period should be determined carefully.

Table 5.2: Yield achieved and required s-TSV count for b17 with different timing budgets.

Timing budget ($c\%$)	Achieved yield	# Inserted s-TSVs
0	NF*	NF
5	NF	NF
10	NF	NF
15	99.80%	567
20	99.80%	491
25	99.80%	421
30	99.80%	375

* NF: No feasible solution found for the target yield

Impact of N_{gs}

N_{gs} is another user-defined parameter in the proposed technique. It denotes the number of inserted s-TSV in each group. In this work, N_{gs} is the same for every group. In order to boost the likelihood of successful repair when multiple TSVs are faulty in the same group, it is beneficial to increase N_{gs}, but it also leads to higher hardware cost. Furthermore, according to Equation (5.12), the delay and area overheads introduced by signal re-routing circuitries (i.e., MUXes) are also increased with higher N_{gs}. Therefore, it is important to explore the trade-off analysis between achieved yield and introduced overheads.

Here the experiment was performed on benchmark b17 with 1,563 f-TSVs, and N_{gs} was varied from 1 to 3 to achieve the same target yield ($Y_{TSV} = 99.99\%$). Three options are given for the TSV defect probability: 0.01, 0.001, and 0.0001 [160], and the defects are assumed to be clustered with $\alpha = 2$. The timing budget is set to $c = 15\%$ relative to LPD of the design.

Table 5.3 illustrates the detailed allocation solutions in terms of s-TSV and MUX counts with varied defect probability. As shown, for a lower defect probability (e.g., $p = 0.0001$), the repair configuration with $N_{gs} = 1$ is a better option due to its lower hardware cost. According to Equation (6.5), the hardware cost not only depends on N_{G_f}, but also increases with N_{gs}. Moreover, due to smaller ΔD_{MUX}, the f-TSV partitioning can be more feasible under the same timing requirement, which leads to a better optimized allocation solution in terms of inserted s-TSV count. Therefore, the repair configuration with lower N_{gs} is a more efficient solution for low defect probability scenarios.

However, for higher defect probabilities (e.g., $p = 0.001$ and $p = 0.01$), the configuration with $N_{gs} = 1$ cannot achieve the target yield. Therefore, it is desirable to assign more s-TSVs for each group in this situation. Moreover, increasing N_{gs} does not always result in higher hardware cost. For example, for the case of $p = 0.001$, the configuration with $N_{gs} = 3$ can guarantee the yield with fewer s-TSVs and MUXes. Although the area of single 4-to-1 MUX is bigger than that for a 3-to-1 MUX, reducing the number of s-TSVs can save far more area compared to the area overhead introduced by MUXes. Therefore, in order to guarantee repairability with as little hardware cost as possible, it is important to select an appropriate value of N_{gs} during s-TSV allocation.

Table 5.3: Trade-off analysis between TSV yield and hardware cost for different values of N_{gs}.

Defect probability	N_{gs}	# Inserted s-TSVs	MUX	
			# MUXes	Type
$p = 0.01$	1	NF*	NF	2-to-1
	2	NF	NF	3-to-1
	3	561	3,687	4-to-1
$p = 0.001$	1	NF	NF	2-to-1
	2	278	3,404	3-to-1
	3	378	3,504	4-to-1
$p = 0.0001$	1	95	3,221	2-to-1
	2	114	3,240	3-to-1
	3	123	3,249	4-to-1

* NF: No feasible solution found for the target yield

5.4.3 Impact of process variation on spare TSV allocation

In order to guarantee the correct operation of a 3D IC, it is necessary to consider process variation during s-TSV allocation. In this work, we consider the impacts of different

process corners on the generated s-TSV allocation solution in terms of achieved yield and inserted s-TSV count.

The experiment was performed on benchmark b17, and N_{gs} is set to 1 for achieving the same target yield ($Y_{TSV} = 99.8\%$). The defects are assumed to be clustered with $\alpha = 2$ and the defect probability is 0.001. Here, three process corners are considered, including *Fast-Fast* (FF), *Typical-Typical* (TT), and *Slow-Slow* (SS). We assume that all the circuit elements in the same 3D IC are manufactured at the same process corner, and the different 3D ICs with varied process corners are designed to achieve the same target yield. The results are presented in Table 5.4. In order to avoid timing errors with the consideration of process variation, all corner cases must be considered, which implies a corresponding increase of clock period. As shown in Table 5.4, the clock period is set to 2.90 ns for the three cases. However, due to limited timing budget, the SS case cannot achieve the target TSV yield. On the other hand, compared to the FF case, excessive s-TSVs are inserted for TT case in order to achieve the same yield value. Consequently, without considering the impact of process variations, the generated s-TSV allocation solution could: i) not achieve the claimed TSV yield, and ii) insert excessive s-TSVs for achieving the target yield.

Table 5.4: Yield achieved and required s-TSV count for b17 with different process corners.

Process corner	CLK (ns)	LPD (ns)	Y_{TSV}	# Inserted s-TSVs
SS	2.90	2.90	96.74%	1,563
TT	2.90	1.64	99.80%	251
FF	2.90	1.04	99.80%	179

5.4.4 Comparison with prior work

Impact of clustering

In the previous technique [174], TSV repair was performed under the assumption of uniform defect distribution. However, as discussed in Section 5.1.4, faulty TSVs tend to be clustered. We next consider different values of the clustering coefficient α to observe the impact of defect clustering on s-TSV allocation.

This experiment was performed on the b17 benchmark. The parameter N_{gs} was set to 1, and the TSV defect probability p was set to 0.001. The timing budget is set to $c = 15\%$ relative to LPD of the design. With different values of α, the proposed technique was applied for different target yields; the results are shown in Table 5.5. Due to the inaccurate estimation of TSV defect distribution, the previous technique [174] (i.e., $\alpha = 0$) cannot achieve the claimed yield with its generated repair solution under the realistic clustered defect distribution. For example, to achieve $Y_{TSV} = 99.8\%$, the technique from [174] underestimates the required number of s-TSVs, which results in solutions that are not feasible.

Table 5.5: Comparison between the number of inserted s-TSVs for uniform and clustered defect distributions.

clustering coefficient	$Y_{TSV} = 100\%$	$Y_{TSV} = 99.8\%$	$Y_{TSV} = 99.6\%$	$Y_{TSV} = 99.4\%$
$\alpha = 0$	417	316	218	142
$\alpha = 1$	NF*	449	304	154
$\alpha = 2$	NF	567	377	216
$\alpha = 3$	NF	NF	NF	297

* NF: No feasible solution found for the target yield

Impact of TSV-induced stress

In prior work, only the re-routing delay overhead was considered during s-TSV allocation [161, 174]. However, TSV-induced stress can also result in significant delay variation, which was ignored in prior work. Moreover, according to the discussion in Section 5.1.3, temperature can impact thermal stress-induced delay variation significantly, which was ignored in [185]. We next investigate the impact of TSV-induced stress on the delay overhead and also the impact of temperature on stress-induced delay variation.

The evaluation of the proposed technique was performed on all six benchmarks, and the generated results are compared to stress-unaware solutions, in which only re-routing delay overhead $\Delta D(f)$ is considered as objective function during the optimization of s-TSV insertion. The target TSV yield $Y_{TSV} = 99.8\%$ and the defect probability is 0.001 with clustering coefficient $\alpha = 2$, and N_{gs} is set to 1. The timing budget is set to $c = 15\%$ relative to LPD of the design. For the thermal-unaware scenario, we assume a uniform temperature distribution with room temperature (i.e., 25 °C) during the estimation of thermal stress-induced delay overhead.

Table 5.6: Comparison between the stress-unaware delay overhead and stress-aware delay overhead.

Benchmark	LPD (ns)	Delay overhead (ps)		
		Stress-unaware technique [174] (S-U)	Proposed technique	
			Thermal-aware (T-A)	Thermal-unaware (T-U)
b17	1.64	256.96	208.28	197.24
b18	1.93	347.02	242.97	225.81
b19	2.15	438.15	304.99	273.05
b20	1.75	227.57	184.32	172.33
b21	1.45	241.43	197.27	186.44
b22	1.89	291.06	268.71	258.93
Δ_{avg}		S-U vs T-A	T-U vs T-A	
		−20.71% (Improvement)	−6.41% (Error)	

According to the results presented in Table 5.6, we can draw the following conclusions. First, without considering the thermal stress-induced delay overhead, the generated allocation solution could result in timing errors. For example, the stress-unaware

allocation solution can increase the LPD for the design b17 by 15.67%, which exceeds the given timing budget (i.e., $c = 15\%$). Next, the stress-aware technique can reduce the delay overhead introduced by s-TSV allocation by 20.71% on average. At last, the thermal-unaware approach underestimates the impact of higher temperature and significant temperature variation in 3D IC, which leads to an infeasible solution under a realistic scenario. The error is 6.41% on average.

5.4.5 Runtime analysis

To obtain the CPU runtime for the proposed framework, the experiment was performed on the benchmarks b17, b18, b19, and b22. By varying the defect probability, target yield, clustering coefficient, and N_{gs}, the average runtime for each benchmark was calculated. This measure consists of two parts: the runtime for global partitioning and the runtime for detailed s-TSV insertion. As illustrated in Table 5.7, both of these steps can be finished within tens of seconds, even for the largest designs.

Table 5.7: The runtime of the proposed methodology for all benchmarks.

Benchmark	Global partition (s)	Detailed insertion (s)	Total (s)
b17	13.45	23.12	36.57
b18	24.71	46.78	71.49
b19	67.25	57.19	124.44
b20	14.41	20.45	34.86
b21	14.77	26.89	41.66
b22	15.33	34.88	50.21

5.4.6 Discussion

Since s-TSV allocation is performed after placement, the reparability of the 3D IC and the quality of the solution in terms of hardware cost and delay overhead are strongly dependent on the existing whitespace distribution. To ensure that a sufficient amount whitespace is available during TSV repair, we can reserve it during floorplanning [186]. This can result in a potential over-design in terms of chip area as the exact required s-TSV count cannot be estimated accurately in the early design stage. However, such an overhead is inevitable unless an iterative design process is applied to increase the whitespace incrementally, which leads to a longer time to market and extra design cost. Moreover, it would be interesting to consider s-TSV insertion in the early stages of design flow for alleviating routing congestion introduced by s-TSV allocation. By utilizing a congestion estimator at placement stage [137], the infeasible s-TSV candidates for each f-TSV group can be screened, and thus the routing problems can be tackled before detailed insertion.

Using KOZ is a conservative way to prevent any logic gates from being impacted by TSV-induced thermal stress [101]. KOZ is nothing but a region around each TSV from which all gates must “keep out” so that they are not influenced by the TSV-induced

stress to some extent. However, combined with the already large TSV size, KOZ can significantly reduce the placement area available for logic gates. Even worse, the KOZ would restrict s-TSV insertion with the given whitespace distribution in a design. This could lead to increased re-routing delay overhead, which cannot be compensated with the reduction of thermal stress-induced delay variation.

5.5 Conclusion of Chapter

Three-dimensional chip stacking with TSVs has gained traction in recent years as a means to continue Moore's law. However, low yield is one of the key obstacles for industry adoption of TSV-based 3D ICs. We have presented a s-TSV allocation technique for reducing the assembly yield loss. The proposed method facilitates to repair non-uniformly placed functional TSVs under clustered defect distributions. We have presented the trade-off analysis among chip yield, hardware cost and timing performance, and demonstrated the proposed technique can ensure high (targeted) yield, which is not feasible with previous methods.

This page would be intentionally left blank.

6 Electromigration Recovery-aware TSV Repair for Reliability Improvement

In Chapter 4, we study the EM degradation issues in P/G TSV. However, EM wear-out mechanism can also impact the lifetime of signal TSVs in 3D ICs. In order to extend EM-related lifetimes of TSVs, a typical solution is to add s-TSVs in the design to repair defective f-TSVs at run-time. To this end, various TSV redundancy allocation techniques and their corresponding repair algorithms have been proposed in the literature [173, 187]. TSV defects induced by EM can be effectively tolerated by in-field reconfigurable repair solutions. However, the transient recovery effect in EM-induced stress evolution was ignored completely in all these existing *ad hoc* methodologies. Here, the "recovery effect" refers to the EM stress relaxation in the interconnect, which occurs when there is no/lower/reverse current passing. Consequently, this effect can be considered as a healing process extending the lifetime of an interconnect as it will take longer time for the stress to reach to the critical threshold for void nucleation [188, 189]. Such phenomena have been observed in many previous experimental works [190, 191]. According to these experiments, this healing process possesses positive temperature dependence and directional property: on the one hand, higher temperatures lead to faster and more complete recovery of EM stress; on the other hand, this recovery phenomenon is more visible when the interconnect is stressed by bi-directional current waveforms compared to unidirectional ones. Therefore, since most of the f-TSVs in 3D ICs experience very high temperatures and carry bi-directional currents [192], they exhibit significant recovery effect, which can be leveraged for EM-related lifetime enhancement.

In this chapter, a recovery-aware proactive TSV repair solution is proposed to enhance the EM-related lifetime reliability of regular f-TSV grids in which TSVs are placed uniformly. In this repair approach, TSV redundancy is used proactively to allow non-faulty f-TSVs to be temporarily deactivated and be able to recover from certain EM wear-out well before failing. To this purpose, the implementation of the proposed methodology consists of two stages:

- Design-time TSV grouping: After identifying the f-TSVs vulnerable to EM failures, we partition them into groups and then assign s-TSV(s) to each group with appropriate placement. This grouping is implemented based on lifetime reliability as well as signal re-routing constraints.
- Run-time TSV repair: In each group, the logic signals carried by the f-TSVs take turn being transmitted through the assigned spare(s), which allows all the TSVs (including the redundant one(s)) to be deactivated on a rotating basis and recover from EM wear-out.

Overall, our contributions are summarized as follows:

- In order to reduce the delay and area overheads introduced by the proposed repair solution, we proposed a set of design-time optimization techniques to determine optimal TSV grouping in the post-placement/pre-routing stage. Moreover, a greedy group-merge algorithm is proposed to further reduce the area overhead and achieve a better trade-off between lifetime reliability and hardware cost.

- A run-time repair scheme is proposed in this work, consisting of a lightweight reconfigurable re-routing mechanism and two effective recovery schedules. In particular, a static scheduling approach is used when the workloads can be estimated *a priori*. However, in order to handle a more realistic operation condition with dynamic workload changes, a dynamic recovery scheduling approach is also proposed in this work.

Our simulation results demonstrate that:

- Applied to 3D benchmark designs, our proactive repair approach increases EM-related lifetime reliability (measured in *Mean-Time-to-Failure* (MTTF)) of the f-TSV network by up to 12X compared to the conventional reactive method [173] with similar overhead.

- The proposed greedy group-merge algorithm can further reduce the area overhead by up to 15% under the constraints of timing performance, and achieve a better trade-off between lifetime reliability and hardware cost.

- By considering the workload changes and the status of circuit aging, the proposed repair approach with dynamic recovery scheduling can achieve a higher MTTF (up to 4.5X) of the TSV grid compared to the static scheduling.

The rest of this chapter is organized as follows. Preliminaries and related prior work are presented in Section 6.1. The motivation and basic idea of the proposed proactive repair approach are presented in Section 6.2. Section 6.3 and Section 6.4 describe the methodology in detail. In Section 6.5, we report simulation results. Finally, conclusions are drawn in Section 6.6.

6.1 Preliminaries and Related Work

6.1.1 Electromigration recovery effect

Until now, a number of previous works have studied EM issues in 3D ICs, and shows that TSVs are susceptible to EM wear-out [173, 187]. Once the EM-induced hydrostatic tensile stress exceeds a critical value, a void would be formed on the line under TSV, which can increase its resistance, causing path delay fault and eventually open/short defect [173]. However, this time-varying stress can be reduced when the current density in the stressing current goes down (or even negative) temporarily, which is the so-called

"EM recovery effect". This recovery effect can be quite significant when the interconnect is stressed by symmetric bi-directional (bipolar) pulse current waveforms. Moreover, temperature can also affect it, and higher temperatures lead to faster recovery. Due to the recovery property, it takes longer time for the EM-induced stress to reach the critical value, and thus results in a longer lifetime of an interconnect [188].

In order to leverage the recovery effect for lifetime reliability improvement at the system-level, an EM recovery model with "two-step" equivalent DC current was proposed in [193], which can consider transient recovery effect for the EM stress evolution using existing simple EM models. The generation of the equivalent DC current can be divided into two steps: First, an arbitrary waveform with time-varying current and temperature stress (as shown in Figure 6.1(a)) is converted to an equivalent square waveform (red dotted line in Figure 6.1(b)) by matching at both highest peak stress and final stress in each period, instead of only matching the end point in the simple "equivalent DC" method (yellow dashed line in Figure 6.1(b)). Afterwards, the generated current is further parameterized in terms of current density, duty cycle, temperature and time period to define the waveform. As shown in Figure 6.2, compared to the conventional equivalent DC method, the proposed technique in [193] has smaller errors in terms of time-to-failure estimation. By using this new recovery-aware EM DC current model, lifetime can be easily computed for a interconnect wire given the stressing current waveforms.

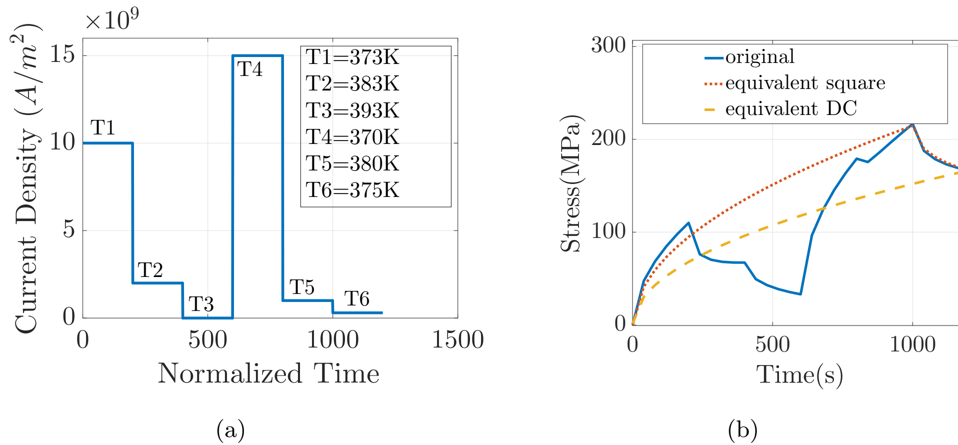


Figure 6.1: (a) Original input driving current density. (b) Calculated EM DC equivalent current density with two different methods.

After void formed, the resistance start increase and this period of time is called growth phase. In the growth phase, we use the average current for simulation since the drift velocity is proportional to current density [189]. As shown in Figure 6.3, the resistance change over time of cases with AC current and its average current are very close. So average current can be applied on the growth phase evaluation.

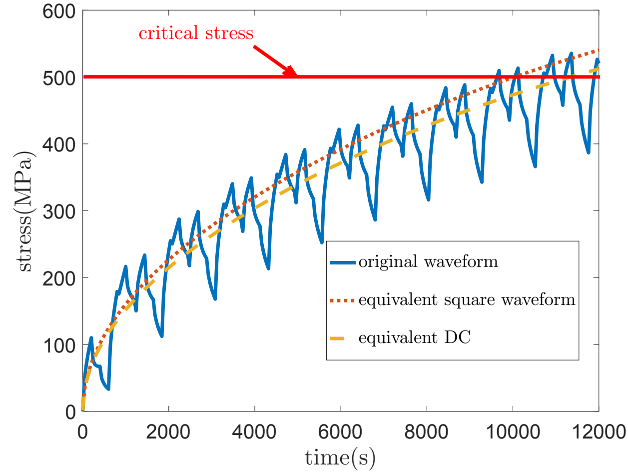


Figure 6.2: Comparing the nucleation time of two different methods and original stress.

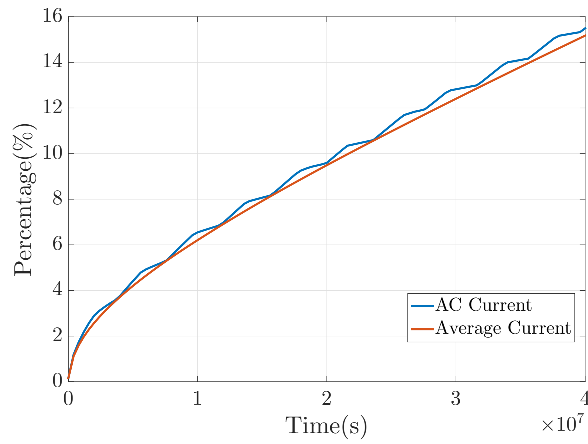


Figure 6.3: Comparing resistance increase with AC current and average current.

6.1.2 Related prior works

A number of s-TSV allocation techniques and their corresponding repair algorithms have been proposed in the literature [159, 173, 194]. However, all of them only target at tolerate manufacturing defects instead of run-time failures (e.g., EM wear-out). To tackle this problem, several in-field repair methodologies were proposed for the post-manufacturing TSV faults [173, 187].

A typical in-field TSV repair scheme is as follows. First, s-TSVs are allocated in the design along with a reconfiguration infrastructure that enables signal re-routing. Afterwards, on-line testing will be triggered periodically or by events. Once a particular f-TSV is detected to be faulty, it would be replaced by a standby s-TSV through the reconfigurable routing network. Therefore, such reactive repair allows as many TSV defects to be tolerated as there are non-faulty s-TSVs. A TSV grid is regarded as being irreparable until the s-TSV resource is exhausted, and its lifetime reliability improvement is highly dependent on the number of allocated s-TSVs.

Due to this "detect-and-replace" scheme, the conventional reactive approach has the following shortcomings:

- For conducting reactive in-field repair, it is imperative to implement an on-chip sensor network in order to test and diagnose faulty f-TSVs, which results in significant hardware cost.
- Ignoring the EM recovery effect, the reactive approach cannot fully utilize the s-TSV resource, which makes the generated repair solution inefficient.

In this chapter, we propose an alternative, more preferred proactive repair approach to address these drawbacks by exploiting the recovery property of EM wear-out.

6.2 Motivation and Basic Idea

As opposed to replacing f-TSVs after they become faulty, the proposed proactive approach allows f-TSVs to recover from EM wear-out before failing. By temporarily deactivating non-faulty f-TSVs, the onset of EM failure can be delayed due to EM recovery effect, which significantly extends TSV lifetime. Therefore, such proactive repair approach has the following advantages over a reactive one:

- Since f-TSVs can recover from EM wear-out before they fail, it is unnecessary to implement the entire on-chip sensor network for TSV defect detection and monitoring, which saves the associated hardware cost.
- In the reactive repair approach, the number of tolerated f-TSV failures is limited by the amount of pre-allocated spares. By contrast, by exploiting the EM recovery effect, proactive approach can extend the lifetimes of multiple f-TSVs even using one single spare, taking full advantage of the limited redundancy resources.

Note that, the conventional reactive repair approach is completely compatible with the proposed one since s-TSVs can still be used if a real f-TSV failure occurs.

The proposed proactive repair approach is based on two consecutive stages. At design-time, the identified EM-vulnerable f-TSVs are partitioned into groups according to their lifetimes, and then s-TSV(s) is (are) subsequently assigned to each group under routing constraints. The corresponding algorithms will be discussed in Section 5.3.6. Afterwards, the assigned s-TSVs are used proactively, which allows partitioned EM-vulnerable f-TSVs in each group to be temporarily deactivated according to different scheduling approaches and recover from EM wear-out before failing. The detailed implementation will be presented in Section 6.4.

6.3 Design-time TSV Grouping

Due to the clustering effects of TSV faults [173], we may run into the situation that some faulty f-TSVs lack TSV redundancy while others have excessive one. Although allocating more s-TSVs can tackle this problem, it also results in significant hardware cost. Therefore, here we propose to:

- Identify the f-TSVs which are vulnerable to EM wear-out at design-time, and limit the use of s-TSVs to them only.
- Adopt the "shared s-TSV" technique [157], which partitions the set of EM-vulnerable f-TSVs into groups and subsequently assign s-TSV(s) to each of them.

Therefore, the "design-time TSV grouping" problem can be consequently divided into the following sub-problems: i) vulnerable f-TSV identification, ii) f-TSV partitioning, and iii) s-TSV assignment.

6.3.1 Electromigration-vulnerable functional TSV identification

As a series system, the EM-related lifetime of a TSV grid is dominated by the f-TSVs which are susceptible to EM failures. Therefore, in order to reduce hardware cost, it is more efficient to provide TSV redundancy to the f-TSVs having lower EM-related lifetimes rather than all of them, and here we use the MTTF of a f-TSV to evaluate its vulnerability. Given a set of representative workloads [195], we can generate the power/thermal characteristics of each f-TSV, and then estimate its MTTF [193]. Note that, since we look into large time scales for EM recovery periods in this work (details in Section 6.4), here a steady-state temperature analysis is sufficient. Then, after comparing with a user-defined threshold value, the f-TSVs with lower MTTF will be identified as EM-vulnerable.

For the f-TSVs with zero or very small timing slacks, a TSV fault is not necessarily a catastrophic open/short defect, but often a timing failure due to EM-induced resistance increase. Therefore, the failure criteria of each TSV (i.e., EM-induced resistance variation) should be considered during timing analysis at design-time, and needs to be determined according to different timing slack of each EM-vulnerable f-TSV. In this

work, a 10% increase from the initial resistance value of the TSV is used as the EM failure criterion, as in [88, 187].

6.3.2 Functional TSV partitioning

After identifying the EM-vulnerable f-TSVs, the next step is to partition them into groups for spare sharing. However, in order to obtain an effective repair solution, it should avoid apportioning the f-TSVs with the lowest MTTF into the same group. Therefore, this problem can be formulated as follows:

- **Input**: i) A set of EM-vulnerable f-TSVs $\mathbf{F} = \{f_i\}$ in which each f-TSV f_i has its MTTF value $\mathrm{LT}(f_i)$; ii) the f-TSV number in each partitioned group N_{gf}.
- **Output**: A set of groups $\mathbf{G} = \{g_j\}$ that partitions $\mathbf{F}$.
- **Constraint**: $\mathbf{F}$ is partitioned into $\lceil |\mathbf{F}| / N_{gf} \rceil$ groups with the most size N_{gf}.
- **Objective**:
$$\text{Minimize} : \max_{\forall g_j \in \mathbf{G}} S(g_j),$$
where $S(g_j) = \sum_{\forall f_i \in g_j} \mathrm{LT}(f_i)$.

Here, the objective can guarantee that the difference of total MTTF value between the maximal and minimal f-TSV groups is minimized, which leads to a uniform partition of vulnerable f-TSVs according to their lifetimes. Then, this f-TSV partitioning problem can be reduced to the Balanced Multi-way Number Partitioning problem [196]. Using the proposed heuristic in [196], we can solve this problem in $O(n\log n)$ time, where $n = |\mathbf{F}|$.

6.3.3 Spare TSV assignment

For each partitioned f-TSV group, we need to subsequently assign s-TSV(s) to provide proactive redundancy, which allows f-TSVs to be temporarily deactivated and recover from EM wear-out. However, in order to maintain the normal operation of circuit, the logic signals carried by the f-TSVs should be capable of being re-routed during field-operation. To this purpose, it is necessary to implement a reconfigurable network for signal re-routing, which inevitably introduces delay overhead. Therefore, the assigned s-TSV(s) for each group should be appropriately chosen among a given set in order to minimize the delay overhead introduced by the in-field repair solution. This issue becomes more critical for high-speed signals (such as memory bus signals), and thus should be accounted properly during TSV repair.

The formal problem statement is as follows:

- **Input**: i) A set of f-TSV groups $\mathbf{G} = \{g_j\}$ that partitions $\mathbf{F} = \{f_i\}$; ii) a set of placed s-TSVs $\mathbf{S} = \{s_k\}$; iii) the assigned s-TSV number N_{gs} for each group.
- **Output**: The mapping between $\mathbf{G}$ and the set of assigned s-TSVs $\mathbf{S}^* \subseteq \mathbf{S}$.

- **Constraint**: The assigned s-TSV number of each group is equal to N_{gs}.
- **Objective**: During in-field repair, the total delay overhead of all groups is minimized. Here the delay overhead of a group is the maximum overhead of all its f-TSVs.

Generally, the delay overhead during repair comes from: i) re-routing logic circuitry and ii) re-routing wire. The first aspect is determined by a given grouping ratio GR $= N_{gf} : N_{gs}$ (details in Section 6.4). Therefore, the objective of this step is to minimize the delay overhead introduced by re-routing wire, and the additional wire length during re-routing is used as a metric to evaluate it, as in [161].

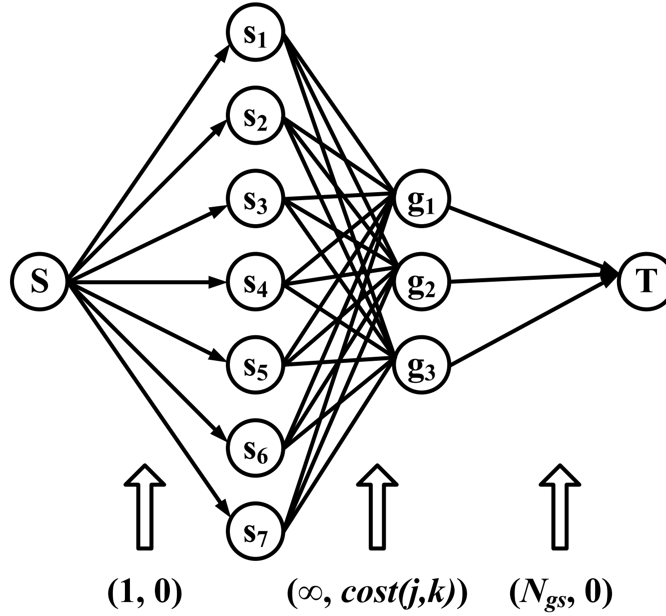


Figure 6.4: Min-cost flow problem for s-TSV assignment.

Here we can formulate the s-TSV assignment as a min-cost flow problem. As illustrated in Figure 6.4, a network $G = (V, E)$ is constructed, whose node set includes all the partitioned f-TSV groups $\{g_j\}$, all the placed s-TSVs $\{s_k\}$, a pseudo source node S, and a pseudo sink node T. There are three kinds of edges in the edge set E, where each edge will be assigned with a (*capacity*, *cost*) pair:

- The source node S has supply of $N_{gs} \times |\mathbf{G}|$, and connect to $|\mathbf{S}|$ s-TSVs $\{s_k\}$. Each edge (S, s_k) has capacity 1 and cost 0.
- There are $|\mathbf{S}| \times |\mathbf{G}|$ edges from the placed s-TSVs $\{s_k\}$ to the partitioned groups $\{g_j\}$. The capacity of edge (s_k, g_j) is infinity. Its cost $cost\,(j,k)$ is the additional wire length during re-routing when assigning s_k to g_j. In other words, $cost\,(j,k) = \max_{\forall f_i \in g_j} L\,(f_i, s_k)$, where $L\,(f_i, s_k)$ is the Euclidean distance between f_i and s_k.

- Every group g_j connects to the sink node T, where each edge (g_j, T) has a capacity of N_{gs} and a cost of 0.

In this min-cost flow problem, the generated solution indicates the optimal assignment of each placed s-TSV to the partitioned group, in the sense of the re-routing additional wire length, and can be solved in polynomial time [197].

6.3.4 Non-uniform TSV grouping

Until now, we assumed that the grouping ratio value GR $= N_{gf} : N_{gs}$ of each TSV group is the same and deterministic. However, in order to fully use the assigned s-TSV, a non-uniform TSV grouping with varying GR for each group seems to be more efficient when targeting a given MTTF. To this purpose, here we propose a greedy group-merge algorithm, as shown in Algorithm 2. Here, the initial f-TSV group $\mathbf{G}_0$ can be obtained by partitioning $\mathbf{F}$ with $N_{gf} = 1$, and then N_{gs} spare(s) is(are) assigned to each group $g_j \in \mathbf{G}_0$ for its MTTF estimation. Note that, since the locations of the assigned spares are determined during s-TSV assignment, their experienced temperatures cannot be obtained at this stage, which are the critical factors affecting EM-related TSV lifetime. In order to guarantee the achieved MTTF of the entire EM-vulnerable f-TSV network can still satisfy the criteria that $\mathrm{MTTF}_{\mathrm{network}} \geq \mathrm{MTTF}_{\mathrm{target}}$ after the subsequent s-TSV assignment stage, here we assume that all the assigned spares experience the highest temperature among all the candidates.

Afterwards, the non-uniform TSV group can be determined iteratively. In each iteration, the two "available" TSV groups with the highest MTTFs are merged together which results in the deletion of N_{gs} spare(s) in the merged group. Note that, here the MTTF of each group is the average lifetime of all f-TSVs in the same group. The term "available" means that the timing slack of each merged EM-vulnerable f-TSV f_i (Slack (f_i)) should satisfy the constraint

$$\text{Slack}\,(f_i) - \Delta D\,(f_i) \geq 0 \tag{6.1}$$

after merging them together. Here $\Delta D\,(f_i)$ is the delay overhead of f_i induced by repair mechanism, which can be introduced by: i) re-routing wire and ii) re-routing logic circuitry. For the first aspect, since the optimal mapping between f-TSVs and s-TSVs has already been determined by the proposed s-TSV assignment in Section 6.3.3, here we only need to make sure to:

- Delete the N_{gs} spare(s) resulting in more re-routing wire-induced delay overhead in each iteration;

- Avoid merging groups which are too far from each other.

Regarding the second aspect, due to the increase delay overhead of re-routing logic circuitry with the group size (discussed in Section 6.4.3), it should be avoided to repartition too many TSVs into the same group.

This iteration can be performed until $\text{MTTF}_{\text{network}}$ decreases to $\text{MTTF}_{\text{target}}$, and then all EM-vulnerable f-TSVs are re-partitioned by the generated $\mathbf{G}'$ during group merging, in which each group has different f-TSV count. Based on this new f-TSV partition, the proposed s-TSV assignment approach is conducted to obtain a non-uniform TSV group, which can achieve the targeted MTTF using less s-TSVs.

Algorithm 2 Iterative method for non-uniform TSV grouping

Input: $\mathbf{G}_0$, $\mathbf{S}$, $\text{MTTF}_{\text{target}}$, N_{gs}, $\Delta D_{\max}$
Output: $\mathbf{G}'$

1: $\mathbf{G}' = \emptyset$;
2: **repeat**
3: sort $\mathbf{G}_0$ based on MTTF in descending order;
4: select the first two available groups in $\mathbf{G}_0$;
5: delete the selected groups in $\mathbf{G}_0$;
6: merge the selected groups together as g_j';
7: delete N_{gs} spare(s) in g_j';
8: re-calculate the MTTF of g_j';
9: $\mathbf{G}_0 = \mathbf{G}_0 \cup \left\{ g_j' \right\}$;
10: update $\text{MTTF}_{\text{network}}$
11: $\mathbf{G}' = \mathbf{G}' \cup \left\{ g_j' \right\}$;
12: **until** $\text{MTTF}_{\text{network}} \geq \text{MTTF}_{\text{target}}$
13: output $\mathbf{G}'$;

6.3.5 Discussion

For generality, all f-TSVs can be divided into three categories according to the signal transportation direction, i.e., the inbound f-TSV, the outbound f-TSV, and the bi-directional f-TSV. In this work, we only target to the uni-directional f-TSVs (i.e., the inbound and outbound ones). Since the opposite direction of current in the bi-directional f-TSVs can compensate EM degradation to some degree, the uni-directional f-TSVs are normally more vulnerable to EM stress compared to the bi-directional ones. In addition, for the f-TSVs with extremely imbalanced bi-directional current, the conventional redundancy approach proposed in [198] can be used to extend their EM-related lifetimes.

6.4 Run-time TSV Repair

After obtaining the TSV groups, the next step is to extend the EM-related lifetime of each EM-vulnerable f-TSV during field-operation. In this work, a run-time TSV repair approach is proposed, consisting of: i) recovery schedule and ii) repair architecture. By the repair architecture, all non-faulty TSVs in each group, which include the assigned spare(s), are allowed to be temporarily deactivated and later reactivated according to

a recovery schedule. This means that the repair architecture provides a configurable routing using the set of TSVs in each group. Therefore, the signals in each group are routed with a subset of TSVs, while the rest can recover from EM wear-out well by exploiting the recovery property. The detailed information is presented in this section.

6.4.1 Recovery schedule

In this work, two recovery scheduling approaches are considered: static scheduling and dynamic scheduling. If the workload can be estimated *a priori* or the changes in the workloads are limited, a static optimization can be utilized for achieving longer TSV lifetime at run-time. However, in many systems, the workload changes dynamically, and the set of running workloads might differ from one chip to other. This means that the amount of EM stress can change over time and from chip to chip. In this case, an on-line approach is desired to manage EM reliability.

Static scheduling

In the static recovery scheduling, EM recovery can occur according to a pre-defined order at design-time. In this work, a periodic recovery schedule is used, in which each TSV is deactivated at regular time intervals, these intervals remain the same over the lifetime operation of the chip. Consequently, according to the grouping ratio, each repair cycle can be split into multiple sub-cycles with the same duration $T_{\text{unit}}^{\text{stat}}$, which is a user-defined parameter. Generally, for a $(N_{gf} : N_{gs})$ group, the repair cycle of each TSV is divided into $(N_{gf} + N_{gs})$ sub-cycles, including active time $T_{\text{active}} = N_{gf}T_{\text{unit}}^{\text{stat}}$ and recovery time $T_{\text{recovery}} = N_{gs}T_{\text{unit}}^{\text{stat}}$. In each sub-cycle, N_{gs} TSV(s) is (are) deactivated for recovery, while the carried signal(s) (if any) will be re-routed through the non-deactivated s-TSV(s).

Dynamic scheduling

The limitation of static scheduling is that it has to be designed for worst case workload over the entire lifetime and all instances of the chip if the running workloads can not be estimated at design-time. Typically, workload characteristics vary dynamically during operational lifetime, and from chip to chip, making it hard to predict workload, and the amount of EM stress, ahead of time. Then a static approach with constant recovery time in each repair cycle would not produce an optimal recovery schedule, tailored to the specific characteristics of the current running workload and the associated EM stress. To overcome this drawback, dynamic management of recovery time is required to achieve higher TSV lifetime and to reduce the reliability challenges caused by EM degradation.

Intuitively, the recovery time in each repair cycle should be dynamically changed according to the degree of EM wear-out of each TSV, which is highly dependent on the running workloads. For a fresh TSV, a larger duration of recovery time in each repair cycle (i.e., $T_{\text{unit}}^{\text{stat}}$) is more desirable, and the slower switching between activated and deactivated modes can reduce the performance degradation introduced by the repair

mechanism. In contrast, with the increased amount of EM wear-out over time, the switching between the two modes should be speeded up (i.e., smaller $T_{\text{unit}}^{\text{stat}}$) in order to achieve a more complete EM recovery during current reversal [188]. In this work, the EM-induced resistance increase is used as the wear-out indicator for each TSV, which can be monitored by the on-chip sensors at run-time [187, 199]. This way, the recovery time of each TSV is almost never constant, and varies with running workload as well as with the EM stress build-up of TSV. Since these sensors measure the change in the TSV resistance due to EM, they act in large time scales [88, 187]. This means that the update in the recovery schedule happens in very coarse time scales (e.g., weeks to months). Therefore, the long-term impact of workload stress can be compensated with this approach.

6.4.2 Repair architecture

In this section, the repair architecture of the proposed run-time TSV repair solution is presented in detail, consisting of: i) a reconfigurable routing network and ii) a recovery selector. For both static and dynamic scheduling, a reconfigurable logic circuitry should be included within each TSV group for signal re-routing. On the other hand, the recovery selector is specifically designed for different scheduling approaches.

Reconfigurable routing network

In order to leverage the recovery effect for EM reliability improvement, each TSV needs to be provided with dedicated shut-off time in the field. Therefore, the signal carried by the deactivated f-TSV should be re-routed to its final destination through another non-deactivated TSV in the same group for maintaining the normal operation. In this work, the assigned s-TSV(s) in each group is (are) served as alternative signal path(s) for the deactivated f-TSV(s), and thus a reconfigurable logic for signal path re-routing should be included within each group.

In order to realize s-TSV sharing and routing reconfiguration, the proposed redundant scheme in [198] is implemented in each group. Here, a $(4:2)$ TSV group is illustrated in Figure 7.3 as an example, in which two dedicated s-TSVs are assigned to a partitioned group consisting of four f-TSVs. As a symmetric scheme, each group needs to be configured both at the receiver and transmitter. To this end, reconfiguration circuitries (i.e., MUXes) are added to the two ends of each TSV, and every single input of the group can be selected and transmitted over the dedicated lines provided by the assigned s-TSVs when its original f-TSV is deactivated. In this way, all TSVs (including the assigned s-TSVs) can operate either in active mode or in recovery mode, and transition between them according to a recovery schedule.

Recovery selector

As shown in Figure 7.3, the selection of deactivated TSVs in each repair sub-cycle is determined by the control signal of each MUX. In other words, the so-called "recovery

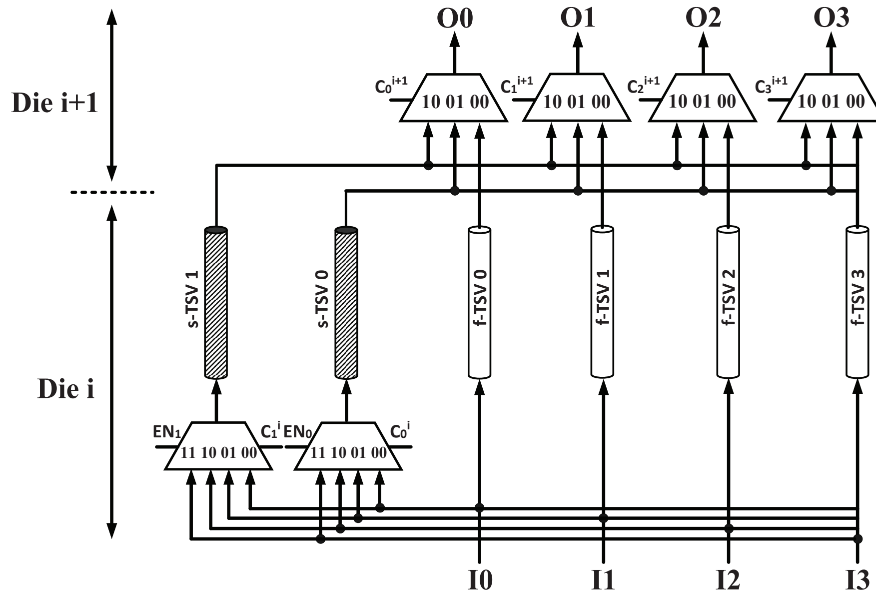


Figure 6.5: Illustration of the reconfigurable routing network for a (4 : 2) TSV group consisting of 4 f-TSVs and 2 s-TSVs.

Table 6.1: The static recovery schedule for a (4 : 2) TSV group. Here, A: active mode, R: recovery mode, fi: f-TSV i, sj: s-TSV j.

TSV	Subcycle_1	Subcycle_2	Subcycle_3	Subcycle_4	Subcycle_5	Subcycle_6
f0	A	**R**	**R**	A	A	A
f1	A	A	**R**	**R**	A	A
f2	A	A	A	**R**	**R**	A
f3	A	A	A	A	**R**	**R**
s0	**R**	A	A	A	A	**R**
s1	**R**	**R**	A	A	A	A
Control signals of MUXes						
EN_0	0	1	1	1	1	0
EN_1	0	0	1	1	1	1
C_0^i	XX	00	00	01	10	XX
C_1^i	XX	XX	01	10	11	11
C_0^{i+1}	00	01	01	00	00	00
C_1^{i+1}	00	00	10	01	00	00
C_2^{i+1}	00	00	00	10	01	00
C_3^{i+1}	00	00	00	00	10	10

schedule" in this context is a set of MUX control signals, which can be changed during field-operation.

The static recovery schedule can be fixed at design-time based on the worst case workload analysis. For example, for the $(4:2)$ group illustrated in Figure 7.3, we list its corresponding static schedule, as shown in Table 6.1. Here, "EN" is the enable pin of all the MUXes at receiver. When EN is set to 0, the MUX is disabled, and the corresponding s-TSV will operate in recovery mode. Otherwise, different f-TSVs will be deactivated according to the control signals. This way, each TSV is deactivated at regular time intervals and provided with uniform recovery time. Moreover, since it is unnecessary to control the MUXes from outside for the static scheduling, the recovery selector here is nothing but a small *Finite-State Machine* (FSM), which generates the control signal for each MUX internally. Since we assume that all TSV groups receive the same control signals, the FSM can be shared between different groups across the entire chip for reducing area overhead.

In order to realize a dynamic recovery schedule, a metric for quantifying the amount of EM stress on a TSV is necessary. As discussed in Section 6.4.1, the EM-induced resistance increase is used in this work to determine the recovery time of each repair cycle. Here, the duration of repair sub-cycle for dynamic repair approach $T_{\text{unit}}^{\text{dyn}}$ can be varied with the EM-induced resistance increase as follows: [1]

$$T_{\text{unit}}^{\text{dyn}} = T_{\text{unit}}^{\text{stat}} \times \left[1 + \alpha\,(M-1)\left(\left\lfloor \frac{\Delta R_{\text{TSV}}^{\text{EM}}\,(t)}{R_{\text{TSV}}^{\text{FC}}} \right\rfloor - \frac{1}{2}\right)\right] \tag{6.2}$$

where $R_{\text{TSV}}^{\text{FC}}$ is the EM failure criterion of TSV (discussed in Section 6.3.1). In this work, the entire range of EM-induced TSV resistance increase between 0 and $R_{\text{TSV}}^{\text{FC}}$ is divided by M pre-defined threshold values into $M-1$ intervals, and the difference of $T_{\text{unit}}^{\text{dyn}}$ between two neighboring intervals is $\alpha \times T_{\text{unit}}^{\text{stat}}$, where α is a user-defined parameter. This way, $T_{\text{unit}}^{\text{dyn}}$ can be varied with the EM-induced resistance increase over time (i.e., $\Delta R_{\text{TSV}}^{\text{EM}}\,(t)$).

In order to obtain $\Delta R_{\text{TSV}}^{\text{EM}}\,(t)$, the on-chip aging sensor [187, 199] can be equipped with f-TSV at design-time. However, as a long-term reliability issue, $\Delta R_{\text{TSV}}^{\text{EM}}\,(t)$ becomes observable only after a long time period (e.g., weeks to months) [88]. In addition, the resolution of such EM-based sensors [187, 199] is limited, and thus only a pretty significant change can be detected by them. As a result, the update of recovery schedule (i.e., the update of $T_{\text{unit}}^{\text{dyn}}$ in each repair cycle) occurs in a large time scale. Since only EM-vulnerable f-TSVs are targeted in this work, all TSVs would be at the similar stage of EM-induced resistance evolution during each update. Therefore, it is unnecessary to monitor each f-TSV for capturing this long-term effect and distinguishing EM stress differences across the f-TSV grid. In contrast, only one sensor is equipped with the entire chip in order to monitor the degree of EM degradation on-line. A few more sensors can be used to avoid variations and improve information accuracy. In addition, for a block-level 3D design, a "one sensor per block" strategy can be adopted to handle the scenario

[1] Here we assume that M is an odd number.

that the utilization rate of each IP block is significantly different. However, for both of the two design granularities, the area overhead introduced by dynamic repair approach is negligibly small due to the limited number of equipped aging sensors.

6.4.3 Overhead analysis

For both the static and dynamic scheduling approaches, the delay and hardware overheads introduced by the proposed repair solution are analyzed as follows.

Delay overhead

For both the static and dynamic scheduling, the delay of control signal generation is negligible (either using small FSM or using off-chip co-processor). Moreover, due to the fact that TSV latency is very small [200], here delay overhead is mainly determined by re-routing wire and reconfiguration circuitries (i.e., MUXes). Although the former one can be minimized by optimal s-TSV assignment based on the given placement of s-TSVs, the inserted MUXes can introduce more significant delay overhead. For a f-TSV f_i in a $(N_{gf} : N_{gs})$ group, its re-routing logic-induced delay overhead is:

$$D\left(f_i\right) = D_{\text{MUX}_{N_{gf}-\text{to}-1}} + D_{\text{MUX}_{(N_{gs}+1)-\text{to}-1}} \tag{6.3}$$

here $D_{\text{MUX}_{N-\text{to}-1}}$ is the propagation delay of a N-to-1 MUX, which can be calculated as:

$$D_{\text{MUX}_{N-\text{to}-1}} = \log_2\left(N\right) D_{\text{MUX}_{\text{2-to-1}}} \tag{6.4}$$

However, using different logic implementation, the actual value of $D_{\text{MUX}_{N-\text{to}-1}}$ can be reduced. According to Equations (6.3) and (6.4), it is desirable to partition f-TSVs into smaller groups (i.e., smaller N_{gf} with respect to the same N_{gs}) in order to reduce the overhead. Note that, for those EM-vulnerable f-TSVs on the critical paths, their timing slacks can be impacted slightly by the added re-routing logic circuitry and its introduced delay overhead. However, this penalty is unavoidable since the EM-induced timing failures can be more severe without the proposed repair solution.

Area overhead

The area overhead can be introduced by: i) reconfiguration routing network and ii) recovery selector. Here we analyze the corresponding overheads for both static and dynamic scheduling approaches as follows.

- **Static scheduling**: For static scheduling approach, recovery schedule is generated by a small FSM internally since it can be fixed at design-time, incurring negligible area overhead. Therefore, the area overhead is dominated by the assigned s-TSVs and added MUXes [173] in the reconfiguration routing network. After implementing TSV grouping with $\text{GR} = N_{gf} : N_{gs}$, the total area overhead of all groups can

be represented as: [2]

$$A = N_{gs} |\mathbf{F}| [A_s/N_{gf} + (2 - 1/N_{gf}) A_{\mathrm{MUX}_{\text{2-to-1}}}] \tag{6.5}$$

where A_s is the area of an s-TSV and $A_{\mathrm{MUX}_{\text{2-to-1}}}$ is the area of a 2-to-1 MUX. Therefore, for a fixed N_{gs}, it is more preferred to partition f-TSVs into larger groups (i.e., larger N_{gf}) to reduce the area overhead.

- **Dynamic scheduling**: For dynamic scheduling approach, the same reconfigurable routing network is used as the static one, and the only difference here is the area overhead introduced by recovery selector (i.e., the aging sensor equipped with TSV). However, since only a few sensors are used for the entire chip (at most one per IP block), the overall overhead is very negligible.

6.5 Numerical Results and Discussions

6.5.1 Experimental setup and implementation flow

For our simulations, six 3D benchmark designs selected from OpenCore benchmark suite [148] were used, including *des_perf-i*, *cf_rca_16-i*, and *cf_fft_256_8-i* ($i = 2, 4$). Here, i is the number of stacked dies in each design. Given the netlist of each design, Cadence SoC Encounter was used to generate layout file using the Nangate 45 nm library [115]. Here, f-TSVs were placed regularly across each die with a 10 μm pitch to form a grid [201], and s-TSVs were placed at the edges of the f-TSV grid with the same pitch [159]. For both f-TSVs and s-TSVs, the total TSV cell size including the KOZ is 8.4 μm, which corresponds to six standard cell rows [202].

Given a grouping ratio $\mathrm{GR} = N_{gf} : N_{gs}$, the proposed TSV grouping technique was conducted on the generated layout files of each design to obtain TSV groups. Afterwards, based on a periodic recovery schedule with a user-defined T_{unit}, the EM model proposed in [193] can be used to estimate the MTTF of each group considering transient recovery effect. To this end, the power/thermal characteristics of each TSV in the group need to be generated. After creating a top-level Verilog netlist for the design, post-synthesis simulation was performed in Modelsim with a testbench containing 10^5 random input vectors. In this way, the switching activity of each f-TSV can be extracted. Moreover, the generated SAIF file was forwarded to Power Compiler in order to obtain the power consumption of each cell. Based on this information and layout files, the experienced temperature of each TSV can be estimated using the 3D Hotspot [149]. In order to obtain the timing slack of each repair-critical TSV, the SPEF file of each die was extracted by performing parasitic extraction after routing. In addition, a top-level Verilog file with the interconnections among dies and a top-level SPEF file with the TSV parasitics were created. Afterwards, all netlist and parasitic information are fed into timing analysis tool (e.g., Synopsys Primetime) to obtain timing slack value for each f-TSV.

[2] Here we assume that $|\mathbf{F}|$ is divisible by N_{gf}.

6.5.2 Impact of design knobs on repair solution

There are two user-defined parameters in the proposed approach, namely T_{unit} and GR. In this section, we investigate their impact on the generated repair solution, and undertake the trade-off analysis between reliability improvement and the corresponding overhead.

Impact of $T_{\text{unit}}^{\text{stat}}$

$T_{\text{unit}}^{\text{stat}}$ is the duration of each sub-cycle in the repair cycle during in-field repair for a static approach. A larger $T_{\text{unit}}^{\text{stat}}$ implies longer recovery time of deactivated TSVs in each repair cycle, but also indicates more EM degradation of the TSVs operating in active mode. As discussed in Section 6.4.1, $T_{\text{unit}}^{\text{stat}}$ is assumed to be constant over time for a static recovery scheduling. Here we present the impact of $T_{\text{unit}}^{\text{stat}}$ on the generated static repair solution in terms of achieved MTTF.

Here the proposed repair solution with static scheduling were conducted on both *des_perf-2* and *cf_fft_256_8-2*, and the grouping ratio GR = 3 : 1. Figure 6.6 illustrates the relationship between the achieved MTTF and $T_{\text{unit}}^{\text{stat}}$. As shown, for both of the two benchmarks, a repair solution with short $T_{\text{unit}}^{\text{stat}}$ (e.g., 10^{-3} s) is incapable of fully exploiting EM recovery effect, which results in an extremely short lifetime. With the increased $T_{\text{unit}}^{\text{stat}}$, the generated repair solution attempts to strike a balance between recovery and degradation in each repair cycle, and achieve it at different $T_{\text{unit}}^{\text{stat}}$ for different benchmarks. However, with the further increase of $T_{\text{unit}}^{\text{stat}}$, the achieved balance becomes disturbed as the EM degradation in each repair cycle can no longer be compensated by recovery effect. As a result, the achieved MTTF becomes lower and saturates finally.

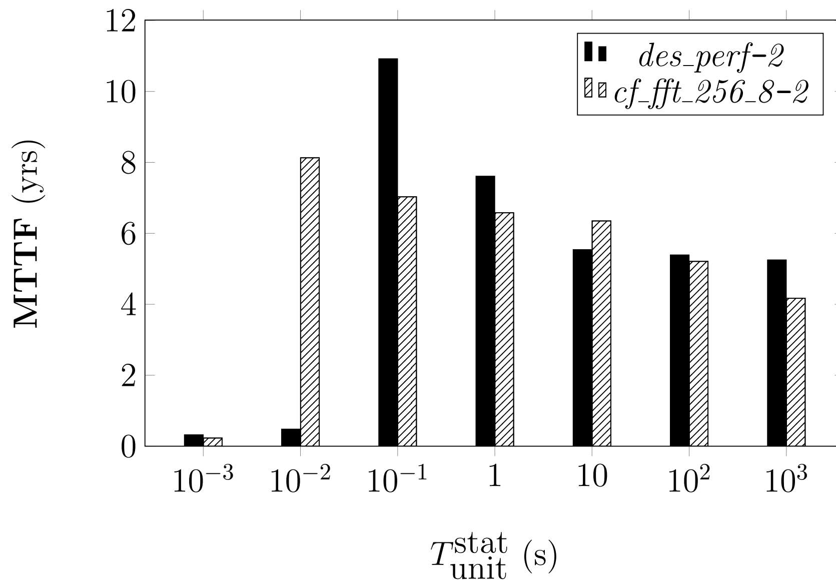


Figure 6.6: The relationship between the achieved MTTF and $T_{\text{unit}}^{\text{stat}}$.

Impact of GR

Grouping ratio GR $= N_{gf} : N_{gs}$ denotes the ratio between the number of f-TSVs and s-TSVs in each group. On the one hand, for a fixed N_{gs}, the partitioning with less N_{gf} leads to better EM recovery in each repair cycle and smaller delay overhead introduced by re-routing logic circuitry (as discussed in Section 6.4), but also results in higher area overhead according to Equation (6.5). On the other hand, for the same GR, the different N_{gs} can also impact the generated repair solution in terms of MTTF and overhead.

In order to evaluate the impact of GR, the proposed repair solution with static recovery scheduling was performed on *des_perf-2*. The duration of each repair sub-cycle $T_{\text{unit}} = 0.1$ s. First, for a fixed $N_{gs} = 1$, N_{gf} was varied from 2 to 4, and the achieved MTTF and the corresponding overhead can be obtained using the proposed approach with different GR. Afterwards, three different cases with GR $= 2:1, 4:2, 6:3$ were considered, in which N_{gs} was varied from 1 to 3 but GR always equals 2. For all the cases, we report the achieved MTTF and the overhead in both area and delay. Here the area overhead is presented in terms of the number of assigned s-TSVs and added MUXes. Since the re-routing wire-induced delay overhead is highly dependent on the given placement of s-TSVs, we only focus on the logic-induced delay overhead here, and report the average value of all the groups.

The results are listed in Table 6.2. Here ΔD is the average delay overhead of all groups in design introduced by repair solution, and ΔA is the percentage of area introduced by repair solution with respect to total cell area including f-TSV and logic cells. As shown, for a fixed N_{gs}, we can achieve a higher MTTF with smaller re-routing logic-induced delay overhead by partitioning less N_{gf} into each group, but also results in larger area overhead. Note that, here reducing the number of s-TSVs can save far more area compared to the area overhead introduced by MUXes. Moreover, for the same GR, assigning more s-TSVs to each group can provide longer recovery time for each repair cycle, which improves lifetime reliability more significantly. However, the penalty is the increased area and delay overheads. Therefore, the desirable GR can be determined by the constraints of timing slacks of repair-critical TSVs (i.e., the target clock period).

Table 6.2: Trade-off analysis between the achieved MTTF and overhead for different grouping ratio.

Grouping ratio	MTTF (yrs)	Overhead	
		ΔD (ps)	ΔA (%)
2:1	11.81	87.717	4.23
3:1	10.92	106.159	2.95
4:1	9.69	116.462	2.24
4:2	12.74	132.260	4.47
6:3	13.67	175.045	5.01

6.5.3 Benefit of non-uniform TSV grouping

As discussed in Section 6.3.4, for a target MTTF, a repair solution using non-uniform TSV grouping can achieve a better trade-off between reliability improvement and area overhead. Therefore, a greedy group-merge algorithm is proposed to divide all EM-vulnerable f-TSVs into several groups with varying N_{gf}. In this section, we show the superiority of the proposed non-uniform grouping approach when targeting the specified MTTF of f-TSV network.

The experiment was conducted on all six benchmark designs as follows. First, the proposed TSV repair solution with static recovery schedule and uniform TSV grouping was performed. Here $\mathrm{GR} = 2:1$ and $T_{\mathrm{unit}} = 0.1$ s. Second, the proposed greedy group-merge algorithm was performed under the timing constraints of EM-vulnerable f-TSVs until the target MTTF ($\mathrm{MTTF}_{\mathrm{target}}$) is achieved. Here $\mathrm{MTTF}_{\mathrm{target}}$ is a user-defined parameter, which can be set according to different application requirements.

The simulation results are presented in Table 6.3. Here ΔD is the delay overhead introduced by repair solution with respect to critical path delay (CPD), and ΔA is the percentage of area introduced by repair solution with respect to total cell area including f-TSV and logic cells. Compared to the repair solution with uniform TSV grouping, the non-uniform solution can achieve the same $\mathrm{MTTF}_{\mathrm{target}}$ (which is less than the optimized one of the uniform solution), but introduce less area overhead. According to our simulation results, the improvement can be achieved by 15.19% in average. The penalty here is the increased delay overhead, which comes from: i) increased re-routing signal path and ii) the more complicated fan-out structure of the MUX circuitry. However, compared to the CPD of each design, this delay overhead introduced by repair solution is very limited. In addition, since the specified timing slack of each f-TSV is considered during non-uniform TSV grouping, the timing correctness can still be guaranteed with the increased delay overhead. Therefore, a better trade-off between timing performance and area overhead can be achieved by the proposed approach for the area-critical systems.

6.5.4 Comparison with conventional reactive repair approach

We compare our proposed proactive repair approach with the conventional reactive one [173], and the results in terms of achieved MTTF are listed in Table 6.4. Here $|\mathbf{F}|$ is the number of EM-vulnerable f-TSVs in each design, and ΔA is the percentage of area introduced by repair solution with respect to total cell area including f-TSV and logic cells. In order to perform a fair comparison, the same power/thermal profiles and recovery-aware EM model are used during TSV lifetime estimation. The only difference between the two schemes is that $T_{\mathrm{unit}}^{\mathrm{stat}} = 0$ for the reactive approach. As shown, the proposed approach with static recovery scheduling can increase MTTF of the TSV grid by up to 12X relative to the reactive method. Moreover, the number of EM-vulnerable f-TSVs $|\mathbf{F}|$ and the area overhead rate ΔA (in terms of percentage of area introduced by s-TSVs and MUXes with respect to total chip area) are also listed in the table. According to our results, the area overhead introduced by the proposed repair solution is pretty small, which can be negligible for a large design. Note that, since here we assume that

Table 6.3: Comparison between the uniform and non-uniform approaches when targeting the same MTTF value of f-TSV network $\text{MTTF}_{\text{target}}$.

Benchmark	CPD (ns)	$\text{MTTF}_{\text{target}}$ (yrs)	$\text{MTTF}_{\text{achieved}}$ (yrs)		ΔD (%)			ΔA (%)		
			Uniform	Non-uniform	Uniform	Non-uniform	Change (%)	Uniform	Non-uniform	Change (%)
des_perf-2	1.75	10	11.81	10.33	7.89	8.81	+11.66	4.23	3.82	-9.69
cf_rca_16-2	2.31	8	9.71	8.14	6.69	7.63	+14.05	1.84	1.60	-13.04
cf_fft_256_8-2	2.92	6	8.51	6.19	5.71	6.35	+11.21	2.23	1.94	-13.00
des_perf-4	1.93	8	9.55	8.21	6.68	7.84	+17.37	6.24	5.09	-18.43
cf_rca_16-4	2.51	7	8.93	7.13	5.89	6.80	+15.45	3.38	2.50	-26.04
cf_fft_256_8-4	3.54	6	8.01	6.23	4.43	5.27	+18.96	2.44	2.06	-15.57
Average							+14.78			-15.19

Table 6.4: Comparison between the proposed proactive approach and the conventional reactive approach [173].

Benchmark	$\|\mathbf{F}\|$	Grouping ratio	ΔA (%)	$\text{MTTF}_{\text{achieved}}$ (yrs)		Improvement (X)
				Proactive	Reactive [173]	
des_perf-2	55	2:1	4.23	11.81	1.62	7.29
		3:1	2.95	10.92	0.98	11.14
		4:1	2.24	9.69	0.87	11.14
cf_rca_16-2	87	2:1	1.84	9.71	3.51	2.78
		3:1	1.25	9.14	3.32	2.75
		4:1	0.97	8.93	2.94	3.04
cf_fft_256_8-2	235	2:1	2.23	8.51	2.21	3.85
		3:1	1.53	8.13	2.04	3.99
		4:1	1.18	7.78	1.78	4.37
des_perf-4	183	2:1	6.24	9.55	1.23	7.76
		3:1	4.26	9.01	0.81	11.12
		4:1	3.30	8.65	0.71	12.18
cf_rca_16-4	218	2:1	3.38	8.93	3.03	2.95
		3:1	2.33	8.21	2.89	2.84
		4:1	1.80	7.99	2.72	2.94
cf_fft_256_8-4	314	2:1	2.44	8.01	2.11	3.80
		3:1	1.68	7.75	1.57	4.94
		4:1	1.30	7.03	1.42	4.95
Average						5.77

the same reconfiguration network is used in both proactive and reactive approaches, the proposed technique does not increase delay and area overheads compared to the baseline, but can achieve a higher MTTF.

6.5.5 Comparison between static and dynamic repair approaches

Until now, we assumed that the executed workloads can be estimated *a priori* or the changes in the workloads are limited over time, and thus a static proactive repair approach with a fixed recovery interval is appropriate in this context. However, in many systems, the workload characteristics vary dynamically in the field, and the set of executed workloads might differ from chip to chip. Therefore, the EM stress build-up of each TSV should be taken into account during recovery scheduling in order to achieve a longer EM-related lifetime, and a dynamic repair approach with varied recovery time is proposed in Section 6.4.1.

In order to evaluate the superiority of the proposed dynamic approach compared to the static one, the experiments were conducted on each benchmark with different $\text{GR} = N_{gf} : N_{gs}$ as follows. For each design, two sets of running workloads are considered (i.e., "normal" and "heavy" workloads), respectively. Here, the "normal" workload is the same with the one used in Section 6.5.4, in which the current density of each f-TSV is extracted from a testbench containing 10^5 random input vectors (as discussed in Section 6.5.1). On the other hand, a "heavy" workload is used to simulate a more

Table 6.5: Comparison between the proposed static and dynamic proactive repair approaches.

Benchmark	Grouping ratio	$\text{MTTF}_{\text{achieved}}$ (yrs)			Improvement (X)
		normal+static	heavy+static	heavy+dynamic	
des_perf-2	2:1	11.81	4.15	17.85	4.30
	3:1	10.92	3.97	14.49	3.65
	4:1	9.69	3.42	11.21	3.28
cf_rca_16-2	2:1	9.71	4.03	9.56	2.37
	3:1	9.14	3.82	8.77	2.30
	4:1	8.93	3.07	8.14	2.65
cf_fft_256_8-2	2:1	8.51	3.81	8.82	2.31
	3:1	8.13	3.06	8.21	2.68
	4:1	7.78	2.87	8.04	2.80
des_perf-4	2:1	9.55	3.25	14.45	4.45
	3:1	9.01	3.03	12.86	4.24
	4:1	8.65	2.85	10.31	3.62
cf_rca_16-4	2:1	8.93	2.94	8.89	3.02
	3:1	8.21	2.77	8.16	2.95
	4:1	7.99	2.41	8.02	3.33
cf_fft_256_8-4	2:1	8.01	2.45	8.11	3.31
	3:1	7.75	2.21	7.94	3.59
	4:1	7.03	2.04	7.41	3.63
Average					3.25

aggressive application scenario, in which the current density of each TSV equals to twice the corresponding value in the "normal" case. Afterwards, three sets of results in terms of the achieved MTTF were extracted and reported in Table 6.5.

First, the static repair approach was applied for different benchmarks with the "normal" running workloads (i.e., "normal+static" case in Table 6.5). Next, considering the "heavy" workload for each benchmark, the static and dynamic repair approaches were used, respectively, and the achieved MTTF values are listed in the forth and fifth columns of Table 6.5 (i.e., "heavy+static" and "heavy+dynamic" cases). According to the results listed in Table 6.5, the achieved MTTF of the "heavy+static" case is significantly reduced compared to the one of the "normal+static" case for each benchmark due to the increased current density and temperature in the "heavy" workloads. This means that under heavy utilization, the static approach may not be able to fully compensate the EM degradation. However, compared to the static repair approach, the dynamic one with varied $T_{\text{unit}}^{\text{dyn}}$ (i.e., the "heavy+dynamic" case) can achieve a higher MTTF, and the improvement can be up to 4.45X according to our simulation results. In this work, $T_{\text{unit}}^{\text{dyn}}$ of each repair cycle can be obtained by Equation (6.2), in which $M = 11$ and $\alpha = 0.1$.

6.5.6 Discussion

In this chapter, we limit our scope only to the signal TSVs. However, as an integral part of PDNs in 3D ICs, P/G TSVs are more susceptible to EM degradation compared to signal TSVs as they experience large amount of unidirectional currents. In [203], a repair solution has been proposed to enhance the EM-related lifetime reliability of P/G

TSV networks by leveraging EM recovery effect.

In this work, we assume that each EM-vulnerable f-TSV and its assigned s-TSV belong to different interconnect trees. Here, the “interconnect tree” is the elemental EM reliability unit, which is a multi-branch interconnect segment consisting of a continuously connected, highly conductive metal lines terminated by diffusion barriers and located within the single level of metallization [204]. However, in the opposite scenario, in which multiple TSVs within the same metal layer are connected, there is a complex correlation between voiding times in the multi-void case due to the interplay between growth kinetics of voids and current redistribution among the TSVs [205]. This more general scenario will be considered as part of future work.

6.6 Conclusion of Chapter

In this chapter, we have proposed a proactive repair approach to combat EM in TSVs by taking use of the EM recovery effect. Applied to 3D benchmark designs, our proactive approach improves the lifetime reliability of TSVs susceptible to EM failure by approximately 12X over the conventional reactive one with less area overhead, even using simple recovery scheduling. Moreover, a dynamic recovery approach with negligible overhead has been proposed to further improve TSV lifetime reliability for a more realistic operation condition.

This page would be intentionally left blank.

7 Cost-efficient Multicast Testing for Interposer-based 3D ICs

In view of increasing wire delay and higher interconnect power consumption, TSVs-based 3D ICs appear to be a promising solution to overcome bottleneck in conventional CMOS scaling. However, volume production and commercial exploitation of 3D ICs are not feasible before pressing concerns about heat dissipation and test cost are adequately addressed. At present, interposer-based 3D ICs (also known as 2.5D ICs) are being advocated as a precursor to 3D ICs [2, 206].

A cross-sectional view of a typical 2.5D IC is presented in Figure 7.1. Instead of vertically stacked dies as in 3D ICs, multiple dies are placed side by side on a silicon interposer [207, 208]. An interposer is a passive device that allows dies to be mounted on it using micro-bumps. Inside the interposer, there are two types of interconnects: the RDL and TSVs. The RDL is a structure of multiple metal layers that provides horizontal D2D interconnects. TSVs, which are connected to C4 bumps, are used for vertical die-to-package interconnects.

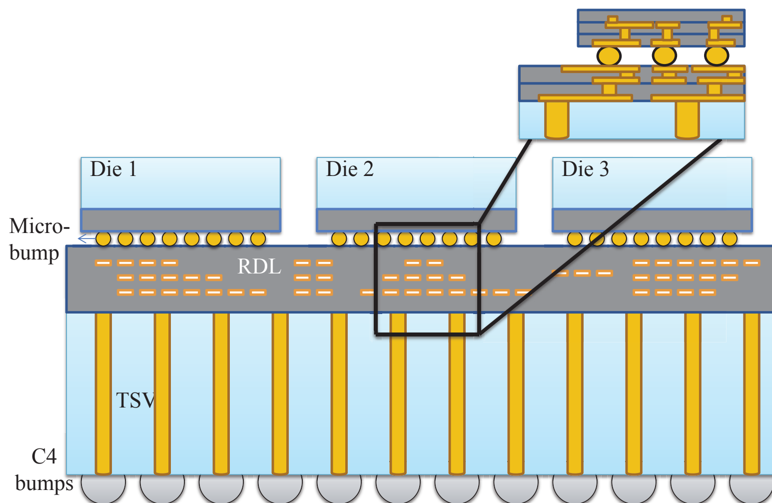


Figure 7.1: Illustration of an interposer-based 3D IC.

Since the size of a micro-bump is small (10 μm in diameter and 40 μm in pitch [209]), a larger number of input and output ports are available for dies in a 2.5D IC. However, the majority of these I/O ports are connected to other dies through horizontal interconnects inside the interposer. The external I/O ports, which are connected to TSVs, are much

fewer in number than the total number of package pins available for the same die in a traditional 2D IC [208]. As a result, the number of test pins available for testing a die in a 2.5D IC is much smaller than that in a 2D package [210].

Based on a design in volume production from an industry collaborator [211, 212], we consider the following example to motivate the test-access challenges in 2.5D integration. Suppose that we have two dies, in which Die I is packaged as a 2D IC consisting of a total of $1,087$ I/O port and Die II is integrated in a 2.5D IC with similar functionality and size as Die I. Since Die II is mounted on the interposer, it has a new I/O interface containing a total of $7,055$ I/O ports. However, the vast majority of these ports are connected to other dies on the interposer (e.g., $6,576$ out of $7,055$ I/O ports in our example), and thus only 479 I/O ports are connected to external I/Os through TSVs, which is much less than the number of package pins available in Die I. In that case, even if all these external I/O ports are available as dedicated test pins, Die II cannot be fully tested with such limited I/O count. This phenomenon can also be observed in another realistic example that has been transitioned to the marketplace—the AMD Fiji chip. In this chip, there are 186k micro-bumps but only 25k C4 bumps [213].

Therefore, with such a limited number of external pins in 2.5D ICs, it is not possible to assign as many dedicated test pins to a die as in 2D ICs. As a result, it is a major challenge to utilize an external tester to apply test patterns generated by an ATPG tool. BIST is therefore a potential solution for testing dies in 2.5D ICs due to the reduced requirement of test pins. A BIST architecture for 2.5D ICs has been presented in [214].

However, the BIST architecture in [214] suffers from two key limitations:

- Since the interposer is a passive device, it cannot support any active logic. Therefore, the BIST architecture and the associated BIST controller must be integrated within each die, and no sharing of the BIST hardware is possible. As a consequence, the area overhead of BIST for a 2.5D IC can potentially be several times larger than that for a corresponding 2D IC. Such increased area overhead is undesirable since it leads to an increase in die area.

- The number of test patterns generated by the BIST architecture of [214] is ten times more than that generated by ATPG. Therefore, an increase in test-application time is inevitable, even with faster in-system and at-speed testing in BIST. In addition, the BIST architecture for die testing in [214] is based on a simple linear-feedback shift-register (LFSR), which cannot target random-pattern-undetectable faults. Consequently, the fault coverage achieved in [214] is unlikely to match the fault coverage provided by ATPG patterns, yet the pattern count is ten times higher. Moreover, the faster test clock in the BIST architecture results in a higher test power consumption, which is also a significant challenge in 2.5D IC testing.

To overcome the above limitations, it is necessary to develop a new methodology for die testing in 2.5D ICs with reduced test cost in terms of test-application time and hardware overhead, but with high fault coverage and less power consumption.

The optimization of test-application time and power consumption has been explored in the context of core-based SoC testing. Given a set of tests for each core and a set

of test resources (e.g., *Test Access Mechanisms* (TAMs)), a test plan can be determined for the application of the tests to the SoC in order to reduce test-application time and power consumption in test mode. Several solutions have been proposed for core-based SoC testing, e.g., [215–224]. Based on TAMs and IEEE Std 1500 core test wrappers, "modular testing" provides a low-cost solution to the SoC test-access problem; a large number of I/O and scan terminals for the embedded cores can be accessed from a few chip pins. However, such techniques are not applicable to 2.5D ICs due to an order of magnitude increase in test challenges and complexity. As explained above, external I/O ports that are connected to TSVs for a SoC die mounted on the interposer are much fewer in count than the total number package pins available for the same SoC in a 2D IC. Therefore, without dedicated testing solutions, post-bond die testing in a 2.5D IC could become an arduous undertaking using the new I/O interface with a reduced number of test pins. To overcome this bottleneck, a post-bond test and DfT strategy for 2.5D ICs was proposed in [225] and further improved in [226], in which the overall test time is minimized under the constraints of the limited available test pins by a parallel TAM optimization approach. However, here the power consumption during 2.5D testing was ignored. Since the dies in 2.5D ICs can provide increased functionality and higher performance [227], the power consumption during testing has also grown dramatically with technology scaling and the relentless increase in design size [228]. Ignoring the test power may result in high IR-drop which may invalidate test [229]. Even worse, heat dissipation during testing in 2.5D ICs becomes much more challenging than conventional 2D ICs as multiple dies generate heat simultaneously but use a common package to dissipate heat [230]. Therefore, test-power issues in 2.5D ICs are far more critical compared to core-based SoCs testing, and existing solution on power-constrained SoC test scheduling are not scalable for 2.5D IC testing.

In this chapter, we propose an efficient multicast test architecture that targets post-bonding die testing in 2.5D ICs. Its main idea is to apply the same test patterns to carefully selected (multiple) dies when they can effectively achieve high fault coverage for these dies within a given test-power budget. In this way, the number of required test pins can be significantly reduced; die testing of 2.5D ICs can therefore be accomplished using ATPG patterns, but with a limited number of test pins. To this end, we propose an efficient test architecture that can enable the application of the same test patterns to multiple dies simultaneously with negligible area overhead. We also present a test-scheduling and optimization technique for identifying groups of dies for multicast, in order to reduce test-application time while satisfying constraints on the power budget and fault coverage.

Our simulation results demonstrate that compared to the BIST technique proposed in [214], the proposed technique reduces test-application time by up to 53.4% for benchmark designs with negligible area overhead. By performing optimal test-scheduling, we achieve even higher fault coverage compared to [214], and the full coverage can be realized using top-off test patterns in addition to multicast test sessions. In addition, compared to the conventional scan-based 2.5D test approach [225], the proposed technique can more effectively reduce test-application under a realistic scenario with limited

test power budget.

The rest of this chapter is organized as follows. Related prior work is presented in Section 7.1. In Section 7.2, we further highlight the motivation for this work with a simple example. Section 7.3 and Section 7.4 present the proposed test architecture and test scheduling techniques, respectively. In Section 7.5, we report simulation results. Finally, conclusions are drawn in Section 7.6.

7.1 Related Prior Work

Interest in 2.5D IC testing has surged in recent years, and several methods have been proposed for die testing in 2.5D ICs [212, 225, 226, 231–235]. In [225], a post-bond test and DfT strategy for 2.5D ICs was proposed, in which functional interconnects in the interposer are re-used as a parallel TAM such that the overall test length of the dies is minimized. Furthermore, the approach was extended in [226] with the concept of multi-visit TAMs, i.e., each parallel TAM allows the same die to be visited multiple times. However, the power consumption during testing was not taken into account in both these methods. In [234], a programmable method for shift-clock stagger assignment was proposed to reduce power supply noise during SoC die testing in 2.5D ICs. Two neighboring cores that share the same power rail do not toggle simultaneously during scan shift. In order to accomplish testing with a small number of test pins, [212] proposed an ExTest scheduling strategy to implement interconnect testing between cores inside a SoC die on the interposer. Using this technique, testing can be accomplished while satisfying the practical constraint that the number of required test pins cannot exceed the number of available pins at the chip level. However, both these methods focus on die-level testing in 2.5D ICs, and they target only a single die. In [235], an industry implementation was introduced targeting at HBM test challenges, in which various solutions were proposed for tests of base dies, dynamic stress testing, TSV testing, Known-Good Stacked Die speed tests, I/O test and characterization, and 2.5D SiP tests. However, the optimizations of test application time and test power consumption were ignored. In conclusion, no method has thus far been proposed to consider 2.5D ICs testing at the system level, in which all dies are tested in the shortest possible time under the given constraints of test power consumption.

Our primary objective is to reduce test-application time for a 2.5D IC with multiple dies mounted on an interposer. As many as four dies on a passive interposer for a 2.5D IC have been reported from industry [207, 209]; the stacking of larger number of dies on the interposer has also been discussed in industry [236, 237]. For such designs, if dies on the interposer are tested serially, the test-application time can be unacceptably high. In order to reduce it, one potential solution is to test all (or several) dies on the interposer concurrently.

Parallel or concurrent testing has been proposed before for SoCs by leveraging *Automatic Test Equipment* (ATE) capabilities [238]. One form of concurrent testing is the broadcast scan technique [239], in which a single tester channel is fed to multiple internal scan chains. However, this scheme and other test techniques based on broadcast

scan require a special ATPG methodology to ensure high fault coverage. To overcome this limitation, [223] proposes a method of sharing test patterns for multiple cores with the assumption that each core has only one scan chain. After finding the best position to merge the test patterns of each core into the test patterns of the core with the longest scan chain, a scan-chain-disable generator is used to retrieve the original test patterns for each core from the shared test patterns. This method requires a complex scan chain-disable generator, which results in high area overhead. [224] proposed an enhanced technique, which is applicable to cores with multiple scan chains and does not require a complicated on-chip decoder. Moreover, test scheduling was proposed to reduce test-application time further under the constraint of TAM width.

Traditional test solutions for core-based SoC testing are ineffective for 2.5D IC testing, as the scale of challenges and complexities (test power, test pin limitations, test time) are an order of magnitude more. Although the broadcast method of [224] is effective for core-based SoC (i.e., a single die in 2.5D IC) testing, it is not applicable to 2.5D IC testing. The lack of effectiveness can be attributed to the two aspects: i) the limited availability of test pins in 2.5D ICs, and ii) the limited power budget for 2.5D IC testing. Generally, two types of testing are involved when the dies in a 2.5D IC are tested: InTest and ExTest [212]. Here *InTest* refers to the testing of the internal logic of all of the cores in each die, while *ExTest* refers to the testing of the interconnects between different cores. Therefore, both test should be carried out for an effective 2.5D IC testing. As a core-based SoC, the hierarchy of a single die is cores $\subset$ SoC and testing targets both the cores and the interconnects among cores. When the similar solutions for core-based SoCs are applied to a 2.5D IC, the entire chip can be viewed as the "SoC" and dies on the interposer are viewed as "cores". In this scenario, interposer interconnects can be tested as the "interconnects" among the cores in the SoC (i.e., ExTest for core-based SoC), and the dies have to be tested as the "cores" in the SoC (i.e., InTest for core-based SoC). However, since our objective of testing in this work is to target the cores inside each die, the hierarchy here is cores $\subset$ SoC die $\subset$ 2.5D IC, which is one level deeper. As a result, a "two-fold" InTest has to be applied in order to test the internal logic of all of the cores in each die on the interposer. Due to the limited test-pin count in 2.5D ICs, the total number of available test pins can support the testing of only one SoC die using the traditional broadcast method. Therefore, in order to realize the "two-fold" InTest, dies in a 2.5D IC have to be tested serially, resulting in extremely high test-application time. Moreover, since all cores are tested concurrently in broadcast testing [240], the power consumption in test mode would be unacceptable if the traditional broadcast method was used for 2.5D ICs.

7.2 Motivation for Proposed Work

In this section, we first use a simple example to show why traditional core-based SoC testing methods are ineffective for 2.5D ICs, and then highlight the motivation for this work. As shown in Figure 7.2(a), three dies in a 2.5D IC are under test, namely Die 1, Die 2, and Die 3. Suppose that each die can support 20 scan chains and includes

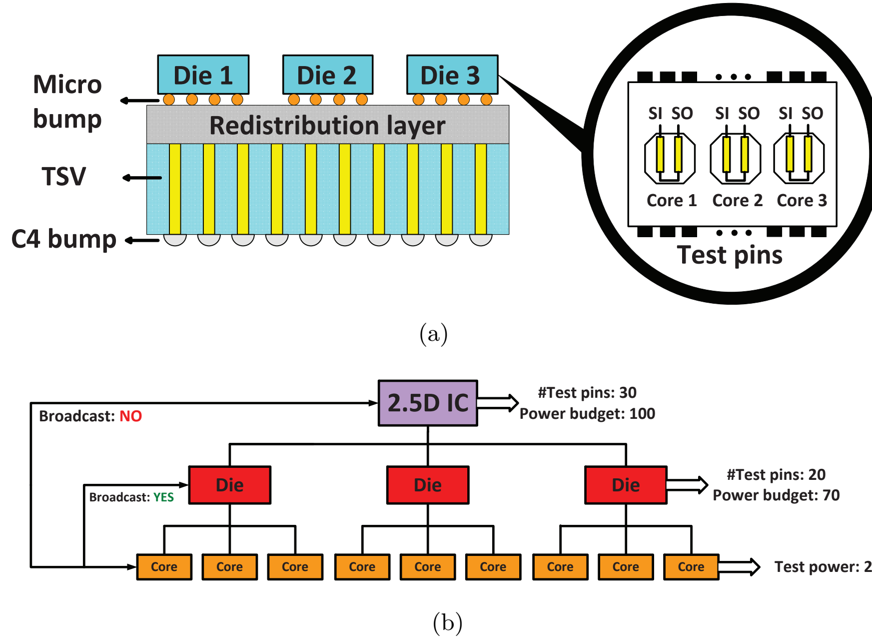


Figure 7.2: (a) Illustration for the motivational example. (b) Hierarchy in 2.5D IC testing: cores ⊂ SoC die ⊂ 2.5D IC.

3 cores. Each core has a full-scan test architecture with multiple scan chains. In this scenario, the traditional broadcast method can be applied *separately (i.e., the dies are tested serially)* to each individual die in the 2.5D IC because each die itself is an SoC design. In other words, the total number of I/O ports (i.e., micro-bumps) in each die can individually support the test of all its cores concurrently. Let us assume that the test-application time for each SoC is 10 arb. unit [1] using the broadcast method. Moreover, let us assume that the power consumption of each core during test mode is 20 arb. unit, and the power budgets for the SoC die and the 2.5D IC are 70 arb. unit and 100 arb. unit, respectively.

In this case, since the given power budget for each SoC die can be satisfied (i.e., $3 \times 20 \leq 70$), the traditional broadcast method can be carried out for each die in which all the cores are tested concurrently. However, this broadcast method cannot be applied *in parallel* to all dies in the 2.5D IC, as shown in Figure 7.2(b). The reasons are as follows:

- First, the number of external I/O ports (i.e., C4 bumps) for a 2.5D IC is much fewer in count than the total number of I/O ports (i.e., micro-bumps) for all dies in the 2.5D IC. For instance, the AMD Fiji chip contains 186k micro-bumps but only 25k C4 bumps [213]. In this case, the available test pins cannot support the concurrent test of the SoC dies using the traditional broadcast method. For instance, in the example 2.5D IC shown in Figure 7.2, three dies are under test

[1] "arb. unit" is the abbreviation for "arbitrary unit".

and each die has 20 scan chains. In order to test the dies in parallel, a total of 60 test pins would be required; however, we assume that there are only 30 test pins available on C4 bumps. If all of the cores are enabled for testing at the same time, although the test-application time can be reduced (10 arb. unit here), the number of required test pins will exceed the number of available test pins. Therefore, broadcast testing cannot be carried out as desired. In this situation, one solution is to test the dies serially, whereby each die can be tested by conventional broadcast method. However, the test-application time would increase significantly. For instance, according to our previous assumption, the total test-application time for the three dies will be 30 arb. unit. When more dies are placed on the interposer, the resulting increase in the test-application time would be unacceptable.

- Second, since the number of cores in a 2.5D IC is equal to the sum of the number of cores in each SoC die in the 2.5D IC, its power budget can be easily exceeded if all dies are tested concurrently using the traditional broadcast method. For instance, using the same assumption as above, if all of the cores are tested concurrently in the 2.5D IC, the total power consumption during test mode is 180 arb. unit (i.e., $3 \times 3 \times 20$), which has exceeded the given power budget for the 2.5D IC (i.e., 100 arb. unit). Therefore, broadcast testing cannot be carried out as desired due to the limited power budget in 2.5D IC testing.

In order to re-use the conventional broadcast approach for core-based SoC testing in 2.5D ICs, the aforementioned two bottlenecks must be overcome. Therefore, the motivation of this work is to propose a more desirable multicast approach for this scenario to handle the trade-off among test-application time, the limited number of available test pins, and the power budget for 2.5D IC testing. Not all dies can be tested in parallel due to the limited test power budget; rather, the dies are scheduled in different test sessions and dies in the same session are tested concurrently. The number and the length of test sessions determine the total test time, and the grouping of the dies into same test session is done based on fault coverage requirements.

In [224], a similar constraint has been considered during the scheduling of broadcast tests in core-based SoCs; it takes into account the limited number of TAM channels. In this scenario, cores are scheduled in different sessions and those in the same session are tested concurrently. However, there are three key differences between our method and [224]:

- Broadcast testing results in the simultaneous toggling of a large number of flip-flops. The increased switching activity leads to higher power consumption in test mode. Power consumption during testing is important since excessive heat dissipation can damage the circuit under test [241, 242]. Therefore, special care must be taken to ensure that the power rating of the circuit is not exceeded during test application. However, the test-scheduling method in [224] is power-unaware, i.e., it ignores the power-consumption limit. In the proposed technique, the power consumption limit for the 2.5D IC is considered during multicast testing.

- In [223] and [224], the precomputed test patterns from core vendors are merged into shared test patterns, which are then broadcast to all the core under test. However, complex hardware is needed to implement the proposed test scheme (either a scan-chain-disable generator for the SoC [223] or a modulo counter for each core [224]), which results in high hardware overhead. The proposed technique uses a simpler design, which results in lower hardware overhead.
- Since the shared test patterns was constructed by merging the precomputed test patterns of all the cores in an SoC from core vendors, the broadcast methods in [223] and [224] do not require fault simulation to verify the fault coverage. However, due to the different test hierarchy in 2.5D ICs, such core-level merging for shared test pattern construction is impractical. In this scenario, the same test patterns should be applied to the dies tested concurrently. However, since the functionalities of different dies on the interposer can be different, the loss of fault coverage for some dies is inevitable during multicast testing. Therefore, it is important to take the potential loss of fault coverage into account, and ensure high fault coverage after multicast testing.

In summary, the proposed multicast test scheme overcomes the limitations of conventional, broadcast-based testing and it is designed to satisfy the following requirements:

- Simple and efficient hardware to implement the proposed test method.
- A test-scheduling strategy that considers power consumption and fault coverage.

7.3 Proposed Test Architecture

7.3.1 Requirement and design

In this subsection, we introduce the new test architecture for multicast testing in 2.5D ICs. The so-called "multicast testing" implies that the same test set (i.e., the multicast test set) are applied to a subset of dies at a time. Here, the *test set* is a set of test patterns which can be generated from ATPG or provided by vendors. In the multicast mode, a group of dies is under test, and their scan chains are enabled to apply the multicast test set. For the other dies not in the multicast group, their scan chains must be disabled so that test set cannot be applied to them. Therefore, the test-application procedure for each die consists of two phases: Test instructions must be first broadcast to each die to activate/deactivate the corresponding test enable signals. Following this step, the multicast test set is applied to these dies whose test enables have been activated. These two phases form one *test session*, and the complete test application for a 2.5D IC consists of multiple test sessions, which ensure that all dies are adequately tested. Note that multicast testing is a general test solution that also includes two special cases: i) serial testing, whereby only one die is tested at a time; ii) broadcast testing, in which all dies are tested simultaneously.

In order to implement the multicast method, two challenges must be addressed during test-architecture design:

- As a passive device, the interposer in 2.5D IC cannot support any active logic. Therefore, the logic implementation of the test architecture has to be realized without any active-logic interposer support.
- The test architecture must include a per-die test enable, which controls whether each die receives the multicast test set to apply it to its internal scan chains or discards it.

To address the above two issues, we propose: i) a "fan-out" structure in the interposer to share the tests from the common test pins between different dies; ii) a multicast-compatible die-level DfT architecture, which requires a multicast-enhanced die wrapper around each die.

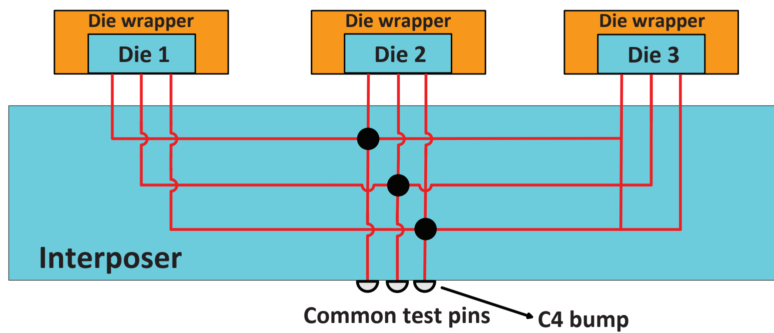
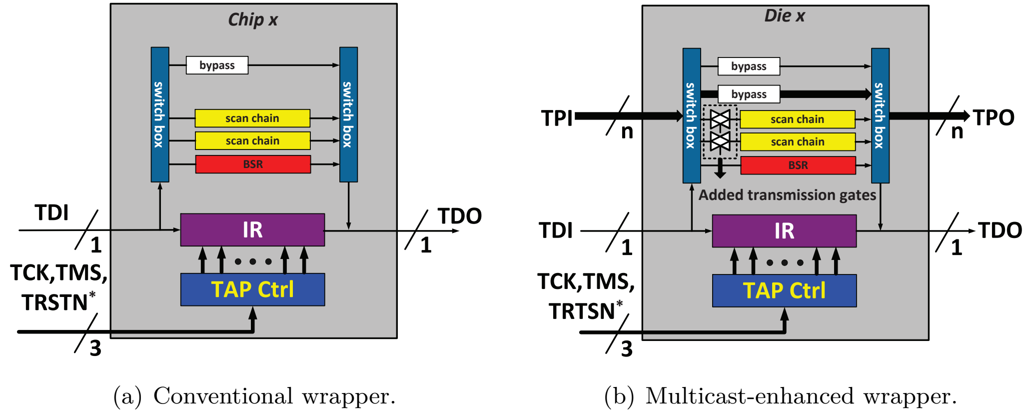


Figure 7.3: Generic test architecture for multicast testing.

The proposed test architecture is illustrated in Figure 7.3. The given k dies (here $k = 3$) are stacked on a silicon interposer and the common test pins are available at the bottom of the interposer through C4 bumps. Each test pin is connected to a fan-out structure so that the tests from one test pin can be applied simultaneously to different dies. For the control of the multicast test architecture, we reuse the IEEE Std. 1149.1 with a slight enhancement, instead of defining an *ad hoc* test control mechanism.

It is well recognized that IEEE Std. 1149.1 standardizes a test wrapper for chips on a PCB [243]. Since dies in a 2.5D IC can be considered similar to chips on a PCB, the IEEE 1149.1 chip wrapper consequently can be used and enhanced to form a die-level wrapper for 2.5D ICs. Figure 7.4(a) presents a conceptual view of a conventional IEEE 1149.1-compliant wrapper. As shown, IEEE 1149.1 wrapper has a two-bit (or optional three-bit) control port, which consists of signals `TCK`, `TMS`, and optionally `TRSTN`*. These control signals can configure a 16-state FSM named *Test Access Port* (TAP) Controller in a mode in which it is ready to receive instructions. Subsequently, the instructions are scanned into the *Instruction Register* (IR) via the `TDI-TDO` interface. Then, the chip is brought into the desired test modes via the control signals and test data is scanned in and out again via the `TDI-TDO` interface. The selected test data register (e.g., a bypass register, a Boundary-Scan Register, or a chip-internal scan chain) depends on the instruction. Note that a single shift register is always selected.



(a) Conventional wrapper.

(b) Multicast-enhanced wrapper.

Figure 7.4: IEEE 1149.1 wrapper: (a) conventional and (b) multicast-enhanced.

However, the conventional IEEE 1149.1 wrapper offers only a serial mechanism. Since it lacks a higher-bandwidth parallel TAM, it is not applicable to our scenario in which test data must be loaded into the internal chains in parallel. Therefore, based on the conventional IEEE 1149.1 wrapper, we propose our multicast-enhanced die-level wrapper, as shown in Figure 7.4(b). This proposed DfT structure differs from the IEEE Std. 1149.1 in the following ways:

- In order to support efficient high-volume testing of the die's internal circuitry, a parallel test port of user-defined width n is provisioned in the proposed wrapper. Here the inputs and outputs of this port are referred to as `TPI` and `TPO`, respectively.

- In order to apply a test set on a die or discard it, each internal scan chain within a die wrapper is equipped with a transmission gate. In this way, when the die is under test, the associated transmission gates are enabled. Otherwise, the transmission gates are disabled so that no test patterns are applied to the die. Note that, although a multiplexer can also be used for this purpose, the transmission gate avoids the unnecessary delay overhead for effective test application time reduction, and introduces less area overhead compared to a multiplexer.

Here the added transmission gates are used to switch between test modes. All the transmission gates within the same die wrapper share the same control signal `MULTICAST_ enable`: when `MULTICAST_enable` is 1, all transmission gates are enabled, which are then used for applying the multicast test set. In order to implement the proposed test method with the TAP controller in IEEE Std. 1149.1, one private instruction needs to be inserted, in addition to the public instructions in IEEE Std. 1149.1, such as BYPASS, IDCODE, and EXTEST. We refer to this instruction as MULTICASTTEST. As shown in Figure 7.5, `MULTICAST_enable` is generated from the decoder, which stores the operational code (opcode) for all instructions, including the newly defined instruction MULTICASTTEST. When the opcode from the IR matches MULTICASTTEST instruction, the decoder sets `MULTICAST_enable` to 1. Otherwise, it sets `MULTICAST_enable`

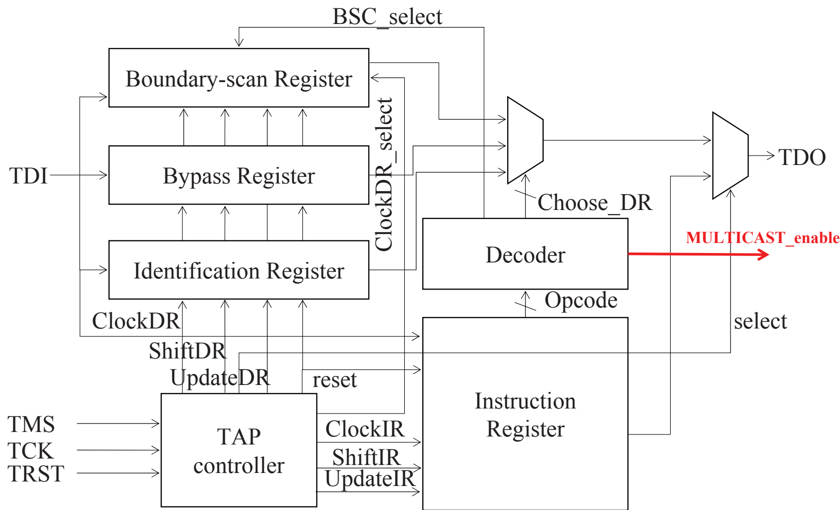


Figure 7.5: Block diagram of the proposed test architecture.

to 0. Note that since this multicast-enhanced die-level wrapper is backward compatible with the conventional IEEE 1149.1 wrapper, conventional testing is not impacted by the proposed multicast test architecture when a die used for a 2.5D IC is reused for other purposes.

We recognize that test access for 3D ICs is currently being standardized through the proposed IEEE P1838 standard [244]. While some aspects of P1838 can be utilized for the proposed test-access architecture, multicast testing of the dies on an interposer does not require test-data transportation vertically up-and-down a 3D stack. Hence we have adopted a simpler test-access architecture based on the IEEE Std. 1149.1.

Note that in this work, without loss of generality, we assume that all dies on the interposer are tested using the same number of internal scan chains. If dies have different scan chain counts, we can implement the proposed method in two ways. In one implementation, we take the maximum number of scan chains for the dies as the number of test channels from the ATE to the dies through the interposer. For dies with fewer scan chains, not all the test chains will be utilized. In the second implementation, the number of tester channels equals the minimum number of scan chains over all the die. The die wrapper is then used to map the smaller number of tester channels to a larger number of scan chains within the die.

The overall optimization and test procedures can therefore be described as follows:

- Perform test scheduling to divide the set of dies in the 2.5D IC into several test sessions.
- Load instructions into the IRs of the different dies.
- After the first test session is completed, load instructions for the dies so that transmission gates for all dies in the next session are enabled.

- Continue this procedure until all test sessions are completed.

7.3.2 Overhead analysis

In this subsection, we analyze the overhead introduced by the new test architecture for multicast testing in 2.5D ICs, which includes: i) timing overhead introduced by the new defined instruction, and ii) area overhead introduced by the proposed test architecture.

Although a wait time is introduced for loading the new instruction, the delivery of this instruction does not take a significant amount of time. The loading of the instruction is accomplished in one IR cycle. In an IR cycle, the controller goes through "Select IR_Scan" (1 cycle), "Capture_IR" (1 cycle), "Shift_IR" (32 cycles), "Exit_IR" (1 cycle) and the "Update_IR" (1 cycle). Since the length of the instruction is 32 bits, the controller stays in the "Shift_IR" state for 32 cycles to shift in the instruction code. Therefore, the total number of clock cycles used to load one instruction is 36 (i.e., $1 + 1 + 32 + 1 + 1$). However, since internal scan chains in a die usually contain hundreds or thousands of scan cells, the total number of clock cycles used to shift in one test pattern can be of the order of hundreds or thousands of cycles. Therefore, the time taken for applying the control data to all the dies is negligibly small, especially in comparison to the volume of multicast test data.

For example, consider a 2.5D IC design crafted using the Opencores benchmarks [2] as dies on the interposer. Three dies are stacked on the interposer: Des_perf, Ethernet, and Rs_dec. The length of the largest internal scan chain in each die is 441, 557, and 361, respectively. Assume that all the three dies are placed in the same test session. Therefore, the same test set is applied to all of them, and the number of the test patterns in the multicast test set is $3,549$. Since the instruction can only be loaded serially via the 1-bit `TDI-TDO` interface, the time taken for applying the control data is 108 cycles (i.e., 36×3, here 3 is the number of dies). However, the time for applying the multicast test set is determined by the die with the largest internal scan (here it is Ethernet), which is equal to $1,976,793$ cycles (i.e., $557 \times 3,549$). Therefore, the timing overhead of applying the control data is only 0.0055%, which is negligible.

The area overhead of the proposed test architecture comes from: i) the added transmission gates within a die wrapper; ii) the fan-out interconnect structure in the interposer. According to our simulation results, the area of each transmission gate is only $2.023\ \mu\text{m}^2$. Therefore, the area overhead of the multicast-enhanced die wrapper (in μm^2) is equal to $2.023 \times N_{\text{scan_chains}}$ compared to the conventional IEEE 1149.1 wrapper. Here $N_{\text{scan_chains}}$ is the number of internal scan chains in each die. In this work, we assume that $N_{\text{scan_chains}}$ is the same for each die, which is equal to 20. Then the area overhead of additional transmission gates for each die is $40.46\ \mu\text{m}^2$. Considering the Ethernet die whose area is $94,835\ \mu\text{m}^2$, the transmission gate area overhead is only 0.043%, which is negligibly small.

In order to realize the fan-out interconnect structure in the interposer, additional TSVs and metal interconnects in RDL are required. The number of the additional

[2] http://opencores.org

TSVs, which is the same as the number of common test pins, is equal to $2 \times N_{\text{scan_chains}}$. Since the common test pins have to provide the channels for both shifting in test patterns and shifting out test responses, their number should be the twice of $N_{\text{scan_chains}}$. Using the same assumption as above, the number of additional TSVs is 40. Assuming that in the realistic scenario, the interposer contains approximately $1,500$ functional TSVs [245], the TSV overhead amounts to only 2.67% in terms of the number of TSVs. Moreover, since the density of functional TSVs in the interposer is low to moderate (due to the limited number of dies on the interposer), the additional TSVs can potentially be accommodated in unused areas of the interposer. Hence they will not increase the area of the interposer, and the TSV overhead can be neglected. For the metal interconnects in the RDL, we assume that they are 45 nm in width, 45 nm in pitch and 100 nm in thickness, which reflect the state-of-the-art interposer technology [246]. In addition, each interconnect is assumed to be $1,700$ μm in length according to the information from our industry collaborator. Therefore, each interconnect would take up 1.53×10^{-4} mm^2 (i.e., $(45+45) \times 10^{-6} \times 1.7$). Assume that three dies are stacked on the interposer and each die is equipped with 20 scan chains. Then the total additional area introduced by the wires in the RDL would be 1.84×10^{-2} mm^2. Note that since the total interposer area is more than 20 mm^2 [246], the additional interconnects only take up less than 0.1% of the total interposer area. As a result, the area overhead of the additional wires in RDL can be neglected. In addition, since the number of additional TSVs and metal interconnects is limited, the testing of interposer itself will not be impacted by the fan-out interconnect structure introduced by the multicast testing architecture. For the sake of clarity, a summary of the above overhead analysis is listed in Table 7.1.

Table 7.1: Summary of timing and area overheads analysis

Time for loading test data (# cycles)	Time for loading control data (# cycles)	Overhead (%)
1,976,793	108	0.0055
Benchmark: an artificial 2.5D design crafted using Des_perf, Ethernet, and Rs_dec		
Area of die (μm^2)	Area of transmission gates (μm^2)	Overhead (%)
94,835	40.4	0.043
Benchmark: Ethernet		
No. of functional TSVs	No. of additional TSVs	Overhead (%)
$\approx 1,500$	40	≈ 2.67
Area of interposer (mm^2)	Area of additional wires (mm^2)	Overhead (%)
≥ 20	1.84×10^{-2}	≤ 0.1%

7.4 Test-scheduling Method

In this section, we present the optimization framework for test scheduling based on the proposed multicast testing procedure and architecture.

7.4.1 Problem statement

Test scheduling in this chapter refers to the problem of determining start times for each scheduled test session and the mapping of dies to test sessions such that the total test-application time is minimized. Throughout this chapter, the term test length will be used as a measure of the time required to perform the corresponding test. We present an ILP model for solving the test-scheduling problem under power consumption and fault coverage constraints. The formal problem statement is as follows:

- **Input**: i) A 2.5D IC under test consisting of a set of SoC dies; ii) a set of multicast test-set candidates; iii) the power budget of each test session during multicast testing; iv) the minimum fault coverage requirement for each die.
- **Output**: The set of scheduled test sessions based on the grouping of dies in the 2.5D IC under test.
- **Constraints**: i) Power constraints: the power consumption for the 2.5D IC during each test session cannot exceed the given power budget; ii) fault coverage constraints: for each die, the achieved fault coverage after multicast testing should not be less than its minimum fault coverage requirement.
- **Objective**: Minimize the sum of the test-application time of all the scheduled test sessions.

Before presenting the ILP formulation in detail, we introduce our notation and clarify the assumptions made for test scheduling.

7.4.2 Notation and assumption

Notation

We use the following symbols to facilitate the discussion.

- $\mathbf{D}$: the set of SoC dies in the 2.5D IC under test, where $|\mathbf{D}| = N_D$;
- d_i: an SoC die ($d_i \in \mathbf{D}$);
- F_{d_i}: the target fault coverage of d_i after multicast testing;
- $\mathbf{C}$: a set of multicast test-set candidates, where $|\mathbf{C}| = N_C$;
- c_j: a multicast test-set candidate ($c_j \in \mathbf{C}$);
- t_{ij}: a test-tuple that describes the application of c_j to d_i with the associated test length, power consumption, and fault coverage corresponding to this application;
- $L_{t_{ij}}$: the test length of the test set c_j in t_{ij};
- $P_{t_{ij}}$: the power consumption of t_{ij} (i.e., the power consumption associated with the application of c_j to d_i);

- $F_{t_{ij}}$: the fault coverage of t_{ij} (i.e., the fault coverage provided by the application of c_j to d_i);
- $\mathbf{T}$: the set of the test-tuples, in which each element corresponds to the application of each test-set candidate in $\mathbf{C}$ to each SoC die in $\mathbf{D}$. Thus $|\mathbf{T}| = N_D \times N_C$;
- $\mathbf{S}$: a set of test sessions;
- s_j: a test session belonging to $\mathbf{S}$ (i.e., $s_j \in \mathbf{S}$), in which all die(s) grouped into this test session is (are) applied the same test set c_j;
- L_{s_j}: the test length of s_j, which is equal to the largest test length among all dies in this test session;
- P_{s_j}: the sum of the power consumption of all dies grouped in s_j;
- x_{ij}: a binary variable indicating whether die d_i is tested in test session s_j: $x_{ij} = 1$ when d_i is tested in s_j, and $x_{ij} = 0$ otherwise;
- $P_{\max}$: the given power budget limit for each test session during multicast testing;
- $F^i_{\min}$: the given minimum fault coverage requirement for die d_i after multicast testing.

We illustrate the above terminology with a simple example. Consider four dies that are stacked on an interposer forming a set of dies $\mathbf{D}$, namely d_1, d_2, d_3 and d_4. In addition, there is a set of test-set candidates $\mathbf{C}$ for multicast testing, which consists of five candidates, namely c_1, c_2, c_3, c_4 and c_5. By applying each candidate in $\mathbf{C}$ to each die belonging to $\mathbf{D}$, we can obtain the Cartesian product of the test sets, and describe it by a set of test-tuples $\mathbf{T}$, whose size is 20 (i.e., 4×5). For example, the test-tuple $t_{12} \in \mathbf{T}$ can be used to describe the application of c_2 to d_1 with the associated $L_{t_{12}}$, $P_{t_{12}}$, and $F_{t_{12}}$. After test scheduling, several die(s) is (are) grouped into the same test session. For instance, d_1 and d_3 are grouped into s_2, which implies that d_1 and d_3 will be tested concurrently using the same test-set candidate c_2. In this case, L_{s_2} and P_{s_2} are determined by the associated power consumption and test length of the test tuples t_{12} and t_{32}. Note that P_{s_2} equals the sum of $P_{t_{12}}$ and $P_{t_{32}}$. If we assume $L_{t_{12}} = 10$ and $L_{t_{32}} = 15$, then $L_{s_2} = L_{t_{32}} = 15$.

Assumptions

We make the following two assumptions in order to simplify the optimization problem:

- Each die is tested only once during multicast testing, which means that each die can only be scheduled in exactly one test session. In a more general scenario, a die can be tested in multiple test sessions with different test sets, which will be considered as part of future work.

- All the dies in the 2.5D IC under test subject to the same minimum fault coverage requirement. In other words, $F^i_{\min} = F_{\min}$ for all dies d_i in the 2.5D IC under test, where $F_{\min}$ is the given common minimum fault coverage requirement for all dies. This assumption does not in any way limit the applicability of our solution. The general case as stated in the problem statement can be handled by including a separate fault coverage constraint for each die.

7.4.3 Integer linear programming formulation

In this subsection, we describe the ILP model for minimizing test-application time under power and fault coverage constraints. Although ILP models are computationally intractable and often not feasible for large problem instances, with a limited number of dies per interposer (for instance, if we consider up to eight dies as a practical scenario), the problem instance here is small enough to be amenable for ILP.

As described in Section 7.4.2, t_{ij} is a tuple to describe the application of test-set candidate c_j to the SoC die d_i with the associated test length $L_{t_{ij}}$ of c_j, the power consumption $P_{t_{ij}}$ during the application, and the fault coverage $F_{t_{ij}}$ provided for d_i. In this case, t_{ij} is a 5-tuple:

$$t_{ij} = \left(d_i, c_j, P_{t_{ij}}, L_{t_{ij}}, F_{t_{ij}}\right) \tag{7.1}$$

The parameters $P_{t_{ij}}$ and $F_{t_{ij}}$ are obtained by power estimation and fault simulation, respectively, and $L_{t_{ij}}$ can be computed as in [247]:

$$L_{t_{ij}} = (1 + ff\,(d_i)) \times N\,(c_j) + ff\,(d_i) \tag{7.2}$$

where $ff\,(d_i)$ is the length of the longest scan chain in d_i, and $N\,(c_j)$ is the number of test patterns in test-set candidate c_j.

Under the first assumption in Section 7.4.2, the aim of test scheduling is to partition the set of dies in the 2.5D IC under test into different test sessions. According to the definition of test session introduced in Section 7.3, the die(s) targeted by the same multicast test-set candidate belong to the same test session, whereby they can be tested concurrently. Let x_{ij} be a binary variable defined as follows: $x_{ij} = 1$ when die d_i is tested in test session s_j, and $x_{ij} = 0$ otherwise. The length of a test session s_j can be expressed as:

$$L_{s_j} = \max\left\{L_{t_{1j}} \cdot x_{1j}, ..., L_{t_{ij}} \cdot x_{ij}, ..., L_{t_{N_Dj}} \cdot x_{N_Dj}\right\} \tag{7.3}$$

Equation (7.3) implies that the test length of a test session equals the maximum test length of all the dies scheduled in this test session.

Each test session s_j must satisfy the following constraints:

$$\begin{aligned} \forall s_j \in \mathbf{S}:\ P_{s_j} &= \sum_{i=1}^{N_D} P_{t_{ij}} \cdot x_{ij} \\ &\le P_{\max} \end{aligned} \tag{7.4}$$

Constraint (7.4) implies that for each test session $s_j \in \mathbf{S}$, the power consumption during it cannot exceed the limited power budget $P_{\max}$.

Moreover, each die in the 2.5D IC under test d_i must satisfy the following constraints:

$$\begin{aligned} \forall d_i \in \mathbf{D}: \ F_{d_i} &= \sum_{j=1}^{N_C} F_{t_{ij}} \cdot x_{ij} \\ &\geq F_{\min} \end{aligned} \tag{7.5}$$

$$\sum_{i=1}^{N_D} x_{ij} = 1, \ j = \{1, 2, ..., N_C\} \tag{7.6}$$

The set of constraints given by (7.5) indicates that for each die $d_i \in \mathbf{D}$, the fault coverage achieved after multicast testing should be not less than the fault coverage limit $F_{\min}$. The constraints (7.6) ensure that each die is tested only once during multicast testing.

Next, we state the objective function of the ILP formulation. The total test length over all test sessions, i.e., $\sum_{j=1}^{N_C} L_{s_j}$ is given by:

$$\sum_{j=1}^{N_C} \max \left\{ L_{t_{1j}} \cdot x_{1j}, ..., L_{t_{ij}} \cdot x_{ij}, ..., L_{t_{N_D j}} \cdot x_{N_D j} \right\} \tag{7.7}$$

where $L_{t_{ij}}$ is the test length corresponding to t_{ij}. Equation (7.7) gives rise to the following objective function:

$$\text{Minimize}: \ \sum_{j=1}^{N_C} L_{s_j} \tag{7.8}$$

The above minmax objective can be simply linearized using the following standard technique associated with ILP modeling:

$$\begin{gathered} y_j \geq L_{t_{ij}} \cdot x_{ij}, 1 \leq i \leq N_D, 1 \leq j \leq N_C \\ \text{Minimize}: \ \sum_{j=1}^{N_C} y_j \end{gathered} \tag{7.9}$$

Therefore, combining Equations (7.4), (7.5), (7.6) and (7.8), we obtain an ILP model to minimize the total test length of all scheduled test sessions, and this model can be solved by an ILP solver. The output of the model indicates: i) which die(s) is (are) scheduled in the same test session and tested concurrently; ii) which multicast test-set candidate is applied to each scheduled test session.

7.5 Simulation Results

7.5.1 Simulation setup

We considered two 2.5D IC designs with different numbers of dies (i.e., 5-die and 8-die cases), which are crafted using the Opencores benchmarks [3] as dies on the interposer. In addition, in order to demonstrate the scalability as well as the applicability of the proposed approach, another larger 2.5D design is also considered (i.e., 4-die case), which consists of four processor-level designs: Aes [4], Leon2 [5]., Leon3mp [5], and Netcard [5]. Details about each design are listed in Table 7.2. We assume that each die has 20 scan chains. The proposed test architecture was synthesized using the 45 nm Synopsys TSMC standard-cell library and Synopsys Design Compiler. Table 7.3 lists the design data for all the benchmarks.

Table 7.2: Design Information for three benchmarks

Die name	4-die case	5-die case	8-die case
Des_perf	✗	✗	✓
Ethernet	✗	✓	✓
Rs_dec	✗	✓	✓
Vga_lcd	✗	✓	✓
Xge_max	✗	✓	✓
Fp_mul	✗	✓	✓
Pci_bridge32	✗	✗	✓
Usb_func	✗	✗	✓
Aes	✓	✗	✗
Leon2	✓	✗	✗
Leon3mp	✓	✗	✗
Netcard	✓	✗	✗

✓: contained in benchmark
✗: non-contained in benchmark

The generated ATPG test set for each benchmark formed the set of multicast test-set candidates. The power consumption was estimated using the Cadence RTL Compiler, and the fault coverage for stuck-at faults was obtained by fault simulation using Mentor Graphics Fastscan. Therefore, for an n-die 2.5D IC, we generated an n×n test-tuple LUT. Based on this LUT, we formulated the ILP model for solving the test-scheduling problem under constraints on power consumption and fault coverage. The ILP solver CPLEX [6] was used to obtain optimal test schedules, and the simulations were performed on a server with four AMD Opteron 6174 processors and 256 GB RAM.

[3] http://opencores.org.
[4] 128-bits advanced encryption standard pipelined cipher, obtained from Opencores.
[5] Obtained from Cobham Gaisler AB, http://www.gaisler.com/.
[6] https://www.ibm.com/bs-en/marketplace/ibm-ilog-cplex#product-header-top

Table 7.3: Design Data for Benchmarks Derived by Commercial Tools

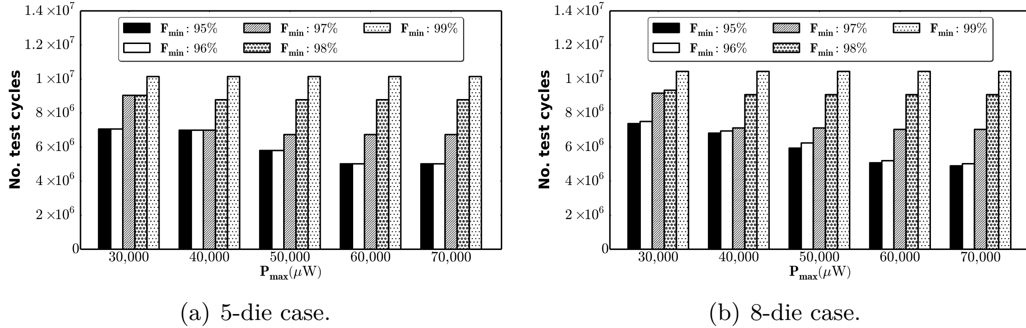
Die name	# Gate	No. Pattern	Length of scan-chain	Fault coverage
Des_perf	98,341	318	441	100%
Ethernet	46,771	3,549	557	99.96%
Rs_dec	44,800	3,061	361	100%
Vga_lcd	124,031	5,347	890	100%
Xge_max	80,833	2,884	709	99.97%
Fp_mul	39,259	953	271	99.9%
Pci_bridge32	16,816	1,078	165	99.99%
Usb_func	12,808	1,397	89	99.98%
Aes	260,541	3,420	539	99.97%
Leon2	590,242	51,749	7,253	99.78%
Leon3mp	299,308	29,017	5,027	99.78%
Netcard	335,586	47,495	4,823	89.58%

7.5.2 Impact of design knobs on test schedule

There are two user-defined parameters in the proposed technique, namely $P_{\max}$ and $F_{\min}$. As expected, both impact the test schedule significantly. In this subsection, we investigate their impact on the generated test schedule, and undertake the trade-off analysis among test-application time, power budget, and testing quality.

Impact of $F_{\min}$

$F_{\min}$ is a crucial parameter in the proposed technique. It specifies the lower bound on the single stuck-at fault coverage that must be achieved for each die. During multicast testing, since the same test patterns are applied to all dies with different functionalities in the same test session, the loss of fault coverage is inevitable. In order to guarantee the high test quality after multicast testing, it is desired to apply appropriate test-set candidate for each scheduled test session.



(a) 5-die case.

(b) 8-die case.

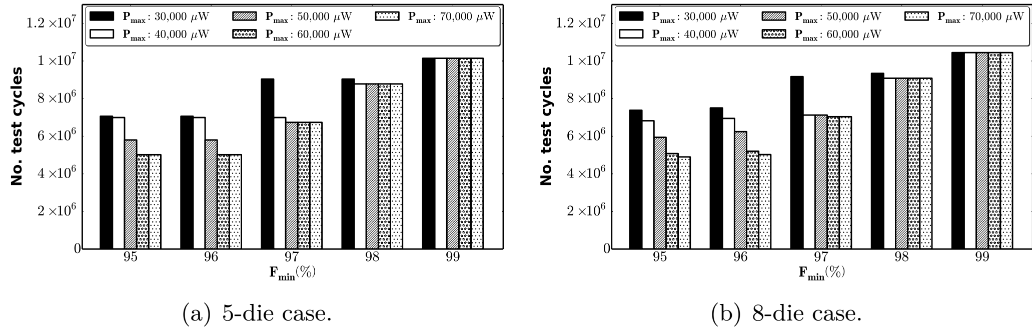
Figure 7.6: The relationship between fault coverage and test-application time with varying power constraints.

Figure 7.6 illustrates the relationship between the test-application time (in terms of the number of cycles) and $F_{\min}$ as $P_{\max}$ is varied. For each value of $P_{\max}$, test scheduling

was performed to achieve different $F_{\min}$ (varied from 95% to 99%). As shown in this figure, for both benchmarks, the test time is lower when a lower minimum fault coverage is set as a constraint for a fixed value of $P_{\max}$. When multicast testing targets lower fault coverage, the test scheduling has more flexibility in satisfying the constraints. For instance, for the 8-die case, our simulation results show that there are six tests for die Fp_mul satisfying the constraints when $F_{\min} = 97\%$ and $P_{\max} = 30,000\ \mu$W. However, for a fixed $P_{\max}$, number of feasible tests is reduced to one when $F_{\min}$ is increased to 99%. Therefore, as the fault coverage constraint is relaxed, more degrees of freedom are available for test scheduling, which leads to solutions with lower test times.

Impact of $P_{\max}$

The power budget $P_{\max}$ is another user-defined parameter in the proposed technique. A higher power budget indicates that more dies can be tested concurrently in each test session, and the test-application time can be reduced significantly. However, an increase in power dissipation during testing will increase the circuit temperature, which results in chip damage, reduced reliability and yield loss. Therefore, it is important to explore the trade-off between test-application time and the power budget.



(a) 5-die case.

(b) 8-die case.

Figure 7.7: The relationship between the power budget and test-application time as $F_{\min}$ is varied.

Figure 7.7 illustrates the relationship between test-application time (in terms of the number of cycles) and $P_{\max}$ as $F_{\min}$ is varied. To achieve each $F_{\min}$, the proposed test scheduling was performed under different power constraints $P_{\max}$ (varied from $30,000\ \mu$W to $70,000\ \mu$W). As shown, for both the 5-die and 8-die cases, with a relaxed power constraint (i.e., high $P_{\max}$), more dies can be tested concurrently due to the higher power budget. Therefore, the number of scheduled test sessions is reduced, which leads to lower test time. Accordingly, there is a trade-off between test-application time and test power budget in multicast test scheduling. The test time can be reduced with a relaxed power constraint, but it results in higher power dissipation during multicast test.

Note that, such test time reduction is more significant when $F_{\min}$ is relatively low (e.g., $F_{\min} = 95\%$). For a higher fault coverage limit, the number of the tests which satisfy

constraints becomes less, as mentioned in Section 7.5.2. Due to the limited solution space, there is less room for further test time improvement using test scheduling. For instance, in order to achieve $F_{\text{min}} = 99\%$, the test-application time of the 8-die case (in terms of the number of cycles) is $10,445,866$ for all scenarios with different power budgets. However, if we reduce the power budget P_{max} to $20,000$ μW, there is no feasible solution for test scheduling, which means that the dies cannot be tested concurrently. As a result, F_{min} plays a more important role in test scheduling for test-time optimization. However, P_{max} is the decisive factor for multicast testing, since it primarily determines which dies in the 2.5D IC can be test concurrently.

7.5.3 Comparison with prior work: BIST-based approach

We compare our results with the corresponding results reported for the BIST method in [214].

Comparison in terms of hardware overhead

In the previous technique [214], the BIST architecture and the associated BIST controller must be implemented within each individual die, and BIST cannot be shared between dies. As a result, the hardware overhead introduced by the BIST architecture for the entire 2.5D IC is considerable. According to the synthesis results shown in [214], the BIST area overhead for each die is $4,077.91$ μm^2 [7]. However, as mentioned in Section 7.3, the only hardware overhead of the proposed architecture is N transmission gates for each die, where N is the number of scan chains per die. For each transmission gate, the area overhead is only 2.023 μm^2 according to our synthesis results. Therefore, the area overhead of the proposed multicast architecture for each die is 40.46 μm^2 for $N = 20$ and grows to only 404.6 μm^2 for $N = 200$, which can be neglected compared to [214].

Comparison in terms of test-application time

To achieve the similar fault coverage, the number of test patterns generated by the BIST architecture is significantly more than those generated by ATPG. Therefore, the increase of test-application time is inevitable even with a faster test clock in BIST architecture. Here we compared our proposed technique to the BIST technique proposed in [214] in terms of test-application time.

Two experiments were performed on both the 5-die and 8-die cases with a fixed power budget of $30,000$ μW to target the fault coverage limit $F_{\text{min}} = 98\%$. Table 7.4 lists the achieved fault coverage by the BIST technique and proposed technique, respectively. According to the results in Table 7.4, we can achieve higher fault coverage for both benchmarks using the proposed technique except Des_perf in the 8-die case. Moreover, compared to the BIST technique proposed in [214], the proposed technique can reduce the total test application time by 50.1% and 53.4%, respectively. We assume here that the test clock in the baseline BIST architecture is the same as the one used in the

[7] This overhead is also estimated using 45 nm Synopsys TSMC standard cell library.

proposed method for evaluating the test-application time. Generally, *Logic Built-in Self-Test* (LBIST) can be operated in multiple frequency domains for different objectives of testing [248]. Moreover, the shift clock frequency in a test-per-scan BIST solution is limited. As explained in [214], an LBIST architecture can operate at slow and fast frequencies to detect stuck-at and transition delay faults, respectively. Therefore, for targeting the stuck-at faults, the test patterns in LBIST have to be applied in system, and the LBIST architecture is operated at the same frequency as one used with the ATE. Therefore, the LBIST frequency is the same as the scan frequency, assumed in our evaluation. In addition, although the BIST architecture can reduce the test-application time further by a faster test clock, the power consumption during testing is increased [249]. In this case, the power consumption constraints are likely to be violated.

Table 7.4: Fault coverage achieved and test-application time for the benchmarks using [214] and the proposed technique, when the test clock frequency is assumed to be 50 MHz.

5-die case		
	[214]	Proposed technique
Die name	Fault coverage	
Ethernet	99.96%	99.96%
Rs_dec	98%	98.02%
Vga_lcd	98%	100%
Xge_max	98%	99.97%
Fp_mul	98%	99.9%
Test-application time (s)	0.3621	0.1808
8-die case		
	[214]	Proposed technique
Die name	Fault coverage	
Des_perf	100%	99.7%
Ethernet	99.96%	99.96%
Rs_dec	98%	98.02%
Vga_lcd	98%	100%
Xge_max	98%	99.97%
Fp_mul	98%	99.9%
Pci_bridge32	96%	99.99%
Usb_func	96%	99.98%
Test-application time (s)	0.4011	0.1868

7.5.4 Comparison with prior work: scan-based approach

In this subsection, we compare our results with the corresponding results reported for the scan-based method in [225]. In [225], a post-bond test and DfT strategy was proposed, which tries to identify an “as-wide-as-possible” set of functional interconnects in the

interposer that can be re-used as parallel TAM to the various dies. Three types of parallel TAM configurations are identified:

- Distribution architecture, in which each die has its own private TAM;
- Daisychain architecture, in which all dies are concatenated by one TAM;
- Hybrid architecture, which is a generalization of distribution and daisychain architectures.

In addition, a search algorithm is proposed in order to identify an optimized TAM configuration that attempts to minimize the overall test length of a 2.5D IC. However, compared to the proposed multi-cast approach in this chapter, there are two major shortcomings of [225], which can be summarized as:

- The external I/Os of the interposer have to be shared by multiple TAMs in parallel in [225], leading to much smaller width of each TAM compared to the multi-cast approach in which the entire I/Os is used by a single TAM. As a result, although all dies can be tested concurrently, the reduction of the overall test length of the dies is still very limited.
- The power consumption during testing is not taken into account in [225]. Consequently, the generated TAM configuration can result in excessive power dissipation during scan testing when all dies are tested concurrently.

Here we compared our proposed technique to the scan-based approach proposed in [225] in terms of achieved fault coverage, test power consumption, and test-application time. Since TAM width is the only objective during TAM configuration optimization in [225], we assume that there is only a single Daisychain TAM in the 2.5D IC achieving the widest width (which equals to the number of entire I/Os) in the generated TAM configuration for the method in [225]. A Daisychain TAM can support both sequential and parallel test schedules, in which one die is tested at a time in the sequential test schedule and all dies are tested simultaneously with their scan chains concatenated in the parallel case. However, for both schedules, the overall test length is approximately the sum of the test lengths of the individual dies [225].

The experiment was performed on the 4-die case, and Table 7.5 lists the achieved fault coverage, power budget per session, and total test-application time by using the proposed technique and the method in [225], respectively. Since the dedicated ATPG test patterns can be applied for each die in the 2.5D IC, the approach proposed in [225] can achieve slightly higher fault coverage for some dies (e.g., Aes and Leon3mp in the 4-die case). However, since the overall test length of the Daisychain TAM is the sum of the test length of each die, the test-application time of the proposed technique in [225] is increased by 24% compared to the proposed approach. In addition, the method in [225] requires higher test power budget per session in order to support the parallel test schedule, which is around twice the value of the proposed approach.

Table 7.5: Fault coverage achieved, required test power budget, and test-application time for the benchmarks using the conventional test approach proposed in [225] and the proposed technique, when the test clock frequency is assumed to be 50 MHz.

4-die case		
	[225]	Proposed technique
Die name	Fault coverage	
Aes	99.97%	99.83%
Leon2	99.78%	99.78%
Leon3mp	99.78%	91.99%
Netcard	89.58%	89.58%
Test-application time (s)	15.0453	12.0903
Test Power budget (mW)	275.19	143.46

7.5.5 Runtime analysis

To evaluate the CPU runtime for the proposed test-scheduling method, the experiments were performed on the 8-die test case with varying value of P_{max} and F_{min}. As illustrated in Table 7.6, all the ILP optimizations can be finished within a few seconds, which indicates the effectiveness of the proposed optimization approach.

Table 7.6: Runtime of the proposed technique with varying constraints of power and fault coverage (unit: second)

P_{max} \ F_{min}	95%	96%	97%	98%	99%
30000 μW	0.05	0.02	0.01	0.01	0.01
40000 μW	0.01	1.48	0.03	2.83	1.37
50000 μW	1.42	0.09	1.39	0.02	0.01
60000 μW	0.04	1.41	0.01	0.01	0.01
70000 μW	0.04	0.04	0.01	0.01	0.01

7.5.6 Discussion

Delay fault testing

In this work, we have targeted only stuck-at faults during multicast testing. In order to handle delay faults (i.e., path-delay and transition faults), at-speed scan testing is needed. In this scenario, test patterns need to launch a transition from a scan cell or primary input and capture the transition at a scan cell or primary output. Therefore, the key to performing such at-speed testing is to generate a pair of rated clock pulses for the launch and capture events. However, this would be complicated for a 2.5D IC design since it involves multiple clocks operating at different frequencies for different dies. In other words, even if the scan shift clock frequencies are the same for all dies,

the capture cycle happens at rated speed, which could be different for different dies. This can cause synchronization issues when the same test is multicasted to multiple dies. Therefore, in order to enable at-speed testing, the 2.5D chip should be designed with an embedded programmable *Phase-Locked Loop* (PLL) that generates sequences of multiple on-chip clock pulses for at-speed testing, as in [250]. Programmability can be achieved using a register-controlled clock-gating circuit that controls the PLL outputs. In addition, in order to synchronize testing multiple dies in parallel, which have their own and different clock frequencies, each die is equipped with First-in-First-out buffers at its test pins [251]. This way, the multicasted test patterns, which arrive simultaneously, can be consumed at different rates according to the dedicated test clock of each die. Using this approach, the proposed multicast test architecture can be extended for delay fault testing in 2.5D ICs.

Compression/Decompression

An important consideration for the proposed multicast testing solution is that some dies mounted on the interposer can potentially use test compression methods in order to reduce test data volume. In this case, the multicast compressed test set targeted one die would achieve very little coverage when applied to other dies in the 2.5D ICs, and the improvement of test-application time introduced by multicast testing would be limited. However, for each die used in a 2.5D design, not all embedded cores are equipped with compression hardware [252, 253]. In addition, due to hardware overhead concerns and the need for diagnosability, some dies may not use compression hardware at all. This situation is common in 2.5D IC because of the heterogeneity of the dies. Therefore, when at least one die in the 2.5D IC is not equipped with compression hardware, the compression hardware has to be disabled for the rest of the dies in broadcast mode. As a result, the cores that are equipped with compression hardware are always designed with features to bypass the compressed test mode [252, 253]; uncompressed test sets are still needed for these cores. On the other hand, for the cores that are not equipped with compression hardware, only uncompressed test sets can be used. Therefore, the proposed multicast test method is generic, which can be applied to any die used in realistic 2.5D ICs.

Moreover, in the scenarios that the compressed test sets can be applied to different targeted dies, the proposed approach can still utilized to reduce test-application time. Due to the limited test-power budget, only a subset of dies can be tested concurrently in each session. Therefore, all the dedicated compressed test sets can be loaded serially for the dies that are supposed to be tested simultaneously. Next, the loaded test sets can be decompressed internally and applied individually at the same time. This way, the test-application time for a session is determined by the longest one of all the dies scheduled in this session. In addition, due to the reduced test data volume in each compressed test set, it take fewer test cycles for loading the test patterns serially than decompressing and applying them. Therefore, the proposed approach can still reduce the test-application time significantly under the given requirement of test power.

Overlapped and non-overlapped testing

In the proposed ILP formulation, we assume that each die in the 2.5D IC under test is tested only once during multicast testing (as shown in Equation (7.6)). However, to achieve higher fault coverage, we can also consider an "overlapped" multicast testing, in which each die can be tested multiple times with different test sets. In this case, the achieved fault coverages of dies can be increased progressively with each scheduled test session. Moreover, since the test length of each test session is determined by the maximum test length of all the dies inside it, such "overlapped" testing can potentially take the same test-application time as the proposed "non-overlapped" testing in which all dies can be tested only once. The penalty of "overlapped" multicast testing is that the power consumption during each test session is likely to increase. This more general scenario will be considered as part of future work.

Compensation for fault coverage loss

Due to the limited number of multicast test-pattern candidates, it is possible that the fault coverage achieved for a die through multicast will be less than the mandated requirement. In this situation, we can increase the fault coverage by applying top-off test patterns to specific cores in a serial manner, which is compatible with the proposed multicast architecture (as mentioned in Section 7.3). Since the loss in fault coverage due to multicast testing is extremely small (as shown in Table 7.4), we can use top-off patterns to achieve full fault coverage for the dies at negligible additional cost.

Compatibility with testing of heterogeneous 2.5D integration

2.5D IC provides a platform to integrate disparate technologies (such as CMOS, memory, sensors, optical interconnects, etc.) on the same interposer, and thus enables heterogeneous integration, in which the mounted dies are almost necessarily not the same size (i.e., the gate counts of different dies have different orders of magnitude). In that case, the proposed test approach in this work is still effective. Suppose that we have three dies mounted on the same interposer as a 2.5D IC, namely Die i, Die ii, and Die iii. Given a power budget M for each test session, the generated test schedule is as follows: Die i and Die ii are tested concurrently in session A, and Die iii is tested individually in session B. Next, let us assume that a significantly larger die Die iv is also mounted on the interposer. In order to realize a feasible test schedule, the power budget of each session has to be increased to N ($N > M$); this power budget is clearly dominated by Die iv. Consequently, Die i, Die ii, and Die iii can be tested concurrently in the same session under the increased test power budget. In order words, the number of test session could be the same, while the overall test length would be dominated by the larger die.

Limitation of proposed approach

Compared to BIST-based test approach, test access using full scan and boundary scan alone requires the use of ATE, which increases the test cost of the proposed approach

and is associated with the tester limitation issues. In addition, it would be challenging to probe the mounted dies in a 2.5D IC because of high density and I/O ports of small dimensions.

7.6 Conclusion of Chapter

The testing of interposer-based 2.5D ICs is an important practical problem today. We have presented a new multicast test architecture that allows multiple dies to be tested concurrently, and described a test scheduling and optimization strategy to determine groups of dies to be tested concurrently and to minimize the overall test-application time. We have presented simulation results to evaluate this technique for two test cases of 5 and 8 dies on an interposer. Detailed test-scheduling results have also been described to highlight the effectiveness of the optimization technique in reducing test time under constraints on test power budget and the fault coverage.

This page would be intentionally left blank.

8 Conclusions

3D ICs, including TSV-based, interposer-based, and monolithic-based schemes, have long been anticipated as the major emerging alternative to classical manufacturing process scaling. Even in the absence of continued device dimensional shrinking, 3D integration technology still has the potential to dramatically enhance chip performance, functionality, and device packing density. Furthermore, the coming wave of devices driving in the *Internet of Things* (IoT) era will require a sophisticated integration with a wide range of functionality, which cannot be economically addressed exclusively with single-die solution. Due to its heterogeneous feature, 3D ICs could become the only candidate that can properly satisfy these requirements. As a result, this technology has drawn a lot of attention from both academia and industry in recent years.

However, the commercial acceptance of 3D ICs still behind the expectation currently, mainly due to new design challenges caused by novel vertical integration. As one of the key enabling technologies in 3D ICs, TSVs can provide vertical signal and power paths between stacked tiers, yet post great threats on the yield, reliability, testability of 3D ICs, which can be summarized as follows:

- **Power integrity:** Smaller footprint combined with multiple tiers in a 3D IC imply severe power integrity issues. This can be further exacerbated by the limited number of available TSV for P/G net routing and EM-induced resistance increase in P/G TSVs, which makes reliable PDN design in 3D ICs highly challenging.

- **TSV reliability and yield:** Various latent TSV defects are introduced by the disruptive manufacturing process involved in TSV fabrication. These defects can be aggravated by the coupling effect between electrical, mechanical, and thermal conditions during in-field operations, which lead to yield loss and reliability degradation, respectively.

- **Test access:** Test access is a major challenge in 3D testing. Since the vertical stacking in 3D ICs introduces a new level of design hierarchy in addition to that in conventional SoCs, new test architectures and approaches are required to provide test access to each tier in a stack. However, the limitations on the number of TSVs and the availability of I/Os only on the tier with the C4 pads give rise to important test-access constraints.

This thesis set out to tackle the aforementioned design and/or test challenges in 3D ICs, and proposes a set of reliability-aware and testability-driven design and optimization techniques for enabling reliable 3D IC designs and enhancing physical design

quality. In this chapter, A brief summary of the main contributions this thesis has developed is presented in Section 8.1, and the future directions arising from this Ph.D. work is discussed in Section 8.2.

8.1 Summary of Contributions

First, a holistic CAD platform for P/G TSV planning has been developed to address the need for reliable PDN design in 3D ICs from the perspectives of power and signal integrities, which can be integrated seamlessly within commercial *Electronic Design Automation* (EDA) development flows. The objective is to handle the trade-off between physical design qualities and hardware design cost by minimizing the inserted TSV (and/or local via) count under the constraints of required design metrics (such as power supply noise and target EM-related lifetime). For TSVs fabricated with different via-first/via-middle/via-last technologies, various optimization approaches are proposed. In Chapter 3, a non-regular P/G TSV planning methodology has been proposed to minimize static power supply noise during power delivery in 3D ICs. The co-optimization of TSV sizing and placement can be finalized in just one step instead of the iterative process in conventional techniques, which reduce computation time significantly. Furthermore, the impact of TSV-induced thermal-mechanical stress on 3D PDN robustness is taken into account during P/G TSV planning. In Chapter 4, a routing-aware P/G TSV planning approach is explored, particularly for TSVs fabricated with via-first technology. To the best of our knowledge, this is the first work considering design routability problems during 3D PDN design. Besides the reduction of placed TSV count, the local vias inside each P/G TSV can also be well-managed to handle the trade-off between routability and reliability of the design.

Second, a comprehensive framework for signal TSV repair has been developed for yield enhancement and reliability improvement in 3D ICs. In general, signal TSVs in 3D ICs suffer from manufacturing defect and latent defects, which can result in yield loss and reliability degradation, respectively. In order to overcome these challenges, a conventional approach is to insert spare elements at design-time and repair faulty f-TSVs in the field. However, due to the huge TSV size and its complicated manufacturing process, the number of inserted s-TSVs should be well-managed. To this end, various spare allocation and TSV repair approaches has been proposed in this thesis. In Chapter 5, a novel s-TSV allocation technique is investigated for improving the assembly yield in TSV fabrication process, which is composed of a light-weight TSV redundancy architecture and two effective repair algorithms. By considering the trade-off between chip yield, hardware cost, and timing performance, this proposed approach can improve chip yield under a more realistic operation scenario. Based on this proposed platform, another TSV repair methodology is proposed to combat EM degradation in TSVs by taking use of its inherent recovery effect. In this approach, TSV redundancy is used proactively, in which non-faulty f-TSVs are allowed to be temporarily deactivated and recover from certain EM wear-out well before failing. This way, the lifetimes of multiple f-TSVs can be extended even using one single spare, taking full advantage of the limited

TSV redundancy resources compared to the conventional reactive manners.

Third, a novel test architecture has been proposed for interposer-based 3D ICs to reduce test-application time and the power consumption during test under the constraints of limited test access. In order to make 3D ICs commercially viable, specified test and DfT solutions are required to address the new challenges for 3D testing. In particular, as 3D stacking introduces a new level of design hierarchy in addition to that in conventional SoCs, new test architectures are needed to provide test access to each die in the stack. In Chapter 7, a multicast test architecture has been proposed for interposer-based 3D ICs (i.e., 2.5D ICs), which targets die testing at post-bonding stage. By applying the same test patterns to carefully selected multiple dies simultaneously with a given test-power budget, the number of required test pins can be significantly reduced. The selection of dies is finalized using a proposed test-scheduling optimization technique.

In conclusion, a set of reliability-aware and testability-driven techniques in the aspects of automating the physical-design process for 3D ICs have been proposed in this dissertation. This process becomes increasingly difficult and demanding compared to well-engineered design automation for their 2D counterparts. On the one hand, classic challenges in 2D ICs such as thermal management, power delivery and placement & routing become more complex in such a scenario. On the other hand, novel 3D-specific design challenges such as the synchronized arrangement of different types of TSVs have undergone similar progress recently. Although more and more sophisticated EDA solutions are being proposed, the design automation for 3D ICs is not yet fully solved. Therefore, the proposed solutions and developed frameworks in this dissertation can act as important building blocks for 3D ICs and push them toward mainstream acceptance in the near future.

8.2 Future Work

The research presented in this thesis seems to have raised more questions that it has answered. There are several lines of research arising from this work which should be pursued, and the 3D integration technology can be further explored in many aspects, such as IoT and neuromorphic engineering.

8.2.1 Leverage 3D integration for internet of things

Although 3D integration has long been anticipated as the major emerging alternative to conventional process down-scaling, the coming wave of devices driving the IoT era will mandate a new set of integration requirements that only 3D ICs can properly satisfy. Most IoT applications require to integrate a variety of functionalities (such as digital, analog, memory, RF modules, MEMS, sensors, and energy harvesters, etc.), which cannot be economically addressed exclusively with single-die solutions. To this end, 3D integration technology can address all this need due to its heterogeneous integration capability. Most industry analysts are anticipating that 3D IC eventually will become the best long-term solution to satisfy increasing requirements in size, power, performance,

and diverse range of IP content in IoT era.

The primary design challenges of an IoT network is security and privacy. As a tremendous number of devices are connected together, an attacker can invade the IoT network through some low-end vulnerable components, which lack security protection as isolated devices previously. Among various threats challenging the security of IoT, the root of trust starts from hardware security. On the other hand, 3D integration technology introduces unique and unexplored challenges on managing hardware security, which are necessary to be identified and mitigated for 3D-based IoT security improvement.

8.2.2 Leverage 3D integration for neuromorphic engineering

Neuromorphic engineering, also known as neuromorphic computing, has emerged in recent years as a complementary architecture to von Neumann architecture. The term "neuromorphic computing" was coined in the late 1980s by Carver Mead [254]. Initially, Mead referred to VLSI containing electronic analog components that mimicked biological neural systems as "neuromorphic" systems. However, in recent times, this term has been used to describe analog, digital, mixed-mode analog/digital VLSI, and software systems that implement models of neural system for perception, motor control, or multi-sensory integration.

Nowadays, the mainstream of neuromorphic hardware design and implementation are based on conventional 2D integration. However, as the neural systems are highly complex, the connectivity densities and massively parallel processed circuital systems are required to engage in highly demanding computational task, which cannot be fulfilled by 2D integration. Instead, 3D integration technology can offer a promising solution to satisfy the aforementioned requirements, which is contemporaneously compatible with current fabrication processes. Actually, 3D integration has been leveraged in neuromorphic computing systems even from the early days of neuromorphic research, especially for pattern recognition and object recognition tasks [255, 256]. In more recent applications, 3D integration has been utilized to stack multiple neuromorphic chips together by using TSVs [257, 258] or micro-bumps [259]. However, the limited number of vertical interconnects achieved by these technologies prevent from the utilization for ultra-dense wires. To this end, 3D MIC seems like an appropriate candidate for neuromorphic computing, yet its implementation has not been demonstrated. Moreover, heat dissipation could be a critical design issue for such 3D MIC-based neuromorphic systems, which need to be mitigated in the future.

Bibliography

[1] G. E. Moore, "Cramming more components onto integrated circuits," *Proceedings of the IEEE*, vol. 86, no. 1, pp. 82–85, 1998.

[2] R. Radojcic, *More-than-Moore 2.5D and 3D SiP Integration.* Springer, 2017.

[3] *International Technology Roadmap for Semiconductors (ITRS'15)*, http://www.itrs.net/.

[4] S. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, D. Finan, P. Iyer, A. Singh, and T. Jacob, "An 80-tile 1.28 TFLOPS network-on-chip in 65nm CMOS," *Proceedings of International Solid State Circuits Conference*, pp. 98–589, 2007.

[5] S. Bell, B. Edwards, J. Amann, R. Conlin, K. Joyce, V. Leung, J. MacKay, M. Reif, L. Bao, and J. Brown, "Tile64-processor: A 64-core soc with mesh interconnect," *Proceedings of International Solid State Circuits Conference*, pp. 88–598, 2008.

[6] R. Kan, T. Tanaka, G. Sugizaki, K. Ishizaka, R. Nishiyama, S. Sakabayashi, Y. Koyanagi, R. Iwatsuki, K. Hayasaka, and T. Uemura, "The 10th generation 16-core sparc64^{TM} processor for mission critical UNIX server," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 1, pp. 32–40, 2014.

[7] W. A. Wulf and S. A. McKee, "Hitting the memory wall: implications of the obvious," *ACM SIGARCH computer architecture news*, vol. 23, no. 1, pp. 20–24, 1995.

[8] H. Jones, "Why migration to 20nm bulk CMOS and 16/14nm FinFETs is not best approach for the semiconductor industry," International Business Strategies, Los Gatos, CA, Tech. Rep., 2014.

[9] J. L. Hennessy and D. A. Patterson, *Computer architecture: A quantitative approach.* Elsevier, 2017.

[10] G.-Q. Zhang and A. V. Roosmalen, *More than moore: Creating High Value Micro/Nanoelectronics Systems.* Springer, 2009.

[11] (2011) The Trend of TSV Packaging. Amkor Technology. [Online]. Available: http://www.sematech.org/meetings/archives/symposia/10202/Keynote-Intro/Lee_The%20Trend%20of%20TSV%20Packaging.pdf

[12] Y.-J. Lee, Y. J. Kim, G. Huang, M. Bakir, Y. Joshi, A. Fedorov, and S. K. Lim, "Co-design of signal, power, and thermal distribution networks for 3D ICs," *Proceedings of Design, Automation, and Test in Europe*, pp. 610–615, 2009.

[13] J. S. Kilby, "Miniaturized electronic circuits," Jun. 23 1964, US Patent 3,138,743.

[14] D. Edenfeld, A. B. Kahng, M. Rodgers, and Y. Zorian, "Technology roadmap for semiconductors," *Computer*, vol. 37, no. 1, pp. 47–56, 2004.

[15] J. W. Joyner, P. Zarkesh-Ha, and J. D. Meindl, "A stochastic global net-length distribution for a three-dimensional system-on-a-chip (3D-SoC)," *Proceedings of International ASIC/SoC Conference*, pp. 147–151, 2001.

[16] J. U. Knickerbocker, P. S. Andry, E. Colgan, B. Dang, T. Dickson, X. Gu, C. Haymes, C. Jahnes, Y. Liu, and J. Maria, “2.5 D and 3D technology challenges and test vehicle demonstrations,” *Electronic Components and Technology Conference*, pp. 1068–1076, 2012.

[17] (2017, Jul. 18) Page 3 Investments Workshop. DARPA. [Online]. Available: https://www.darpa.mil/attachments/Page3WorkshopDay1.pdf

[18] X. Dong, X. Wu, G. Sun, Y. Xie, H. Li, and Y. Chen, “Circuit and microarchitecture evaluation of 3D stacking magnetic RAM (MRAM) as a universal memory replacement,” *Proceedings of ACM/IEEE Design Automation Conference*, pp. 554–559, 2008.

[19] X. Wu, J. Li, L. Zhang, E. Speight, R. Rajamony, and Y. Xie, “Hybrid cache architecture with disparate memory technologies,” *Proceedings of IEEE International Symposium on Computer Architecture*, pp. 34–45, 2009.

[20] G. Sun, X. Dong, Y. Xie, J. Li, and Y. Chen, “A novel architecture of the 3D stacked MRAM L2 cache for CMPs,” *Proceedings of IEEE International Symposium on Computer Architecture*, pp. 239–249, 2009.

[21] D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N. P. Jouppi, M. Fiorentino, A. Davis, N. Binkert, R. G. Beausoleil, and J. H. Ahn, “Corona: System implications of emerging nanophotonic technology,” *Proceedings of IEEE International Symposium on Computer Architecture*, pp. 153–164, 2008.

[22] K. H. Lu, S.-K. Ryu, Q. Zhao, X. Zhang, J. Im, R. Huang, and P. S. Ho, “Thermal stress induced delamination of through silicon vias in 3-D interconnects,” *Electronic Components and Technology Conference*, pp. 40–45, 2010.

[23] A. Todri, S. Kundu, P. Girard, A. Bosio, L. Dilillo, and A. Virazel, “A study of tapered 3-D TSVs for power and thermal integrity,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 2, pp. 306–319, 2013.

[24] C. S. Selvanayagam, J. H. Lau, X. Zhang, S. K. W. Seah, K. Vaidyanathan, and T. C. Chai, “Nonlinear thermal stress/strain analyses of copper filled TSV (through silicon via) and their flip-chip microbumps,” *IEEE Transactions on Advanced Packaging*, vol. 32, no. 4, pp. 720–728, 2009.

[25] E. Eid, T. Lacrevaz, S. de Rivaz, C. Bermond, B. Fléchet, F. Calmon, C. Gontrand, A. Farcy, L. Cadix, and P. Ancey, “Predictive high frequency effects of substrate coupling in 3D integrated circuits stacking,” *Proceedings of IEEE International Conference on 3D System Integration*, pp. 1–6, 2009.

[26] M. Rousseau, M.-A. Jaud, P. Leduc, A. Farcy, and A. Marty, “Impact of substrate coupling induced by 3D-IC architecture on advanced CMOS technology,” *Proceedings of European Conference on Microelectronics and Packaging*, pp. 1–5, 2009.

[27] Z. Xu, A. Beece, D. Zhang, Q. Chen, K.-N. Chen, K. Rose, and J.-Q. Lu, “Crosstalk evaluation, suppression and modeling in 3D through-strata-via (TSV) network,” *Proceedings of IEEE International Conference on 3D System Integration*, pp. 1–8, 2010.

[28] C. Liu, T. Song, J. Cho, J. Kim, J. Kim, and S. K. Lim, “Full-chip TSV-to-TSV coupling analysis and optimization in 3D IC,” *Proceedings of ACM/IEEE Design Automation Conference*, pp. 783–788, 2011.

[29] S. Lee, G. Kim, J. Kim, T. Song, J. Lee, H. Lee, K. Park, and J. Kim, “Modeling and analysis of die-to-die vertical coupling in 3-D IC,” *Proceedings of Electronics Packaging Technology Conference*, pp. 707–711, 2009.

[30] K. Koo, M. Kim, J. J. Kim, J. Kim, and J. Kim, "Vertical noise coupling from on-chip switching-mode power supply in a mixed-signal stacked 3-D IC," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 3, no. 3, pp. 476–488, 2013.

[31] C. Zhang and G. Sun, "Fabrication cost analysis for 2D, 2.5D, and 3D IC designs," *Proceedings of IEEE International Conference on 3D System Integration*, pp. 1–4, 2012.

[32] D. Velenis, E. J. Marinissen, and E. Beyne, "Cost effectiveness of 3D integration options," *Proceedings of IEEE International Conference on 3D System Integration*, pp. 1–6, 2010.

[33] J. H. Lau, "TSV manufacturing yield and hidden costs for 3D IC integration," *Electronic Components and Technology Conference*, pp. 1031–1042, 2010.

[34] Q. Xu, L. Jiang, H. Li, and B. Eklow, "Yield enhancement for 3D-stacked ICs: Recent advances and challenges," *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 731–737, 2012.

[35] N. Miyakawa, "A 3D prototyping chip based on a wafer-level stacking technology," *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 416–420, 2009.

[36] M. Rovitto, "Electromigration Reliability Issue in Interconnects for Three-Dimensional Integration Technologies," Ph.D. dissertation, Vienna University of Technology, 2016.

[37] J. H. Lau, "Overview and outlook of three-dimensional integrated circuit packaging, three-dimensional Si integration, and three-dimensional integrated circuit integration," *Journal of Electronic Packaging*, vol. 136, no. 4, p. 040801, 2014.

[38] D. B. Tuckerman, L. O. Bauer, N. E. Brathwaite, J. Demmin, K. Flatow, R. Hsu, P. Kim, C. M. Lin, K. Lin, and S. Nguyen, "Laminated memory: a new 3-dimensional packaging technology for MCMs," *Proceedings of IEEE Multi-Chip Module Conference*, pp. 58–63, 1994.

[39] E. Beyne, "3D system integration technologies," *Proceedings of IEEE International Symposium on VLSI Technology, Systems, and Applications*, pp. 1–9, 2006.

[40] (2015, Feb. 23) 3D IC packaging & 3D IC integration. Lau, J. H. [Online]. Available: http://s3.amazonaws.com/sdieee/1817-SanDiegoCPMTDL_Lau_advancedpackaging.pdf

[41] S. Mick, L. Luo, J. Wilson, and P. Franzon, "Buried bump and AC coupled interconnection technology," *IEEE Transactions on Advanced Packaging*, vol. 27, no. 1, pp. 121–125, 2004.

[42] K. Kanda, D. D. Antono, K. Ishida, H. Kawaguchi, T. Kuroda, and T. Sakurai, "1.27 Gb/s/pin 3mW/pin wireless superconnect (WSC) interface scheme," *Proceedings of International Solid State Circuits Conference*, pp. 186–187, 2003.

[43] N. Miura and T. Kuroda, "A 1Tb/s 3W inductive-coupling transceiver chip," *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 92–93, 2007.

[44] J. Xu, S. Mick, J. Wilson, L. Luo, K. Chandrasekar, E. Erickson, and P. D. Franzon, "AC coupled interconnect for dense 3-D ICs," *IEEE Transactions on Nuclear Science*, vol. 51, no. 5, pp. 2156–2160, 2004.

[45] Y. Xie, G. H. Loh, B. Black, and K. Bernstein, "Design space exploration for 3D architectures," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 2, no. 2, pp. 65–103, 2006.

[46] Y. Xie, "Processor architecture design using 3D integration technology," *Proceedings of IEEE International Conference on VLSI Design*, pp. 446–451, 2010.

[47] J. Fan and C. S. Tan, "Low temperature wafer-level metal thermo-compression bonding technology for 3D integration," in *Metallurgy-Advances in Materials and Processes*, 2012.

[48] T. Fukushima, Y. Yamada, H. Kikuchi, and M. Koyanagi, "New three-dimensional integration technology using self-assembly technique," *Proceedings of IEEE International Electron Devices Meeting*, pp. 348–351, 2005.

[49] G. Hellings, M. Scholz, M. Detalle, D. Velenis, M. de Potter de ten Broeck, C. Roda Neve, Y. Li, S. Van Huylenbroek, S.-H. Chen, E.-J. Marinissen *et al.*, "Active-lite interposer for 2.5 & 3D integration," *Proceedings of IEEE International Symposium on VLSI Technology*, pp. T222–T223, 2015.

[50] "Samsung Starts Mass Producing Industry's First 3D TSV Technology Based DDR4 Modules for Enterprise Servers," http://www.samsung.com/semiconductor/about-us/news/13602, accessed: 2017-10-27.

[51] "Micron Reinvents DRAM Memory," http://www.linleygroup.com/newsletters/newsletter_detail.php?num=4744&year=2011&tag=3, accessed: 2017-10-27.

[52] "Micron ships Hybrid Memory Cube that boosts DRAM 15X," https://www.computerworld.com/article/2485092/data-center/micron-ships-hybrid-memory-cube-that-boosts-dram-15x.html, accessed: 2017-10-27.

[53] S. Graham, "Development of hybrid memory cube," *3D IC panel discussion, IMAPS 46th International Symposium on Microelectronics*, 2013.

[54] T. Yoshida, M. Hondou, T. Tabata, R. Kan, N. Kiyota, H. Kojima, K. Hosoe, and H. Okano, "Sparc64 XIfx: Fujitsu's Next-Generation Processor for High-Performance Computing," *IEEE Micro*, vol. 35, no. 2, pp. 6–14, 2015.

[55] "Intel's Knights Landing Leverages Technology Found in Micron's HMC Devices," https://www.micron.com/products/hybrid-memory-cube/high-performance-on-package-memory, accessed: 2017-10-27.

[56] C.-C. Lee, C. Hung, C. Cheung, P.-F. Yang, C.-L. Kao, D.-L. Chen, M.-K. Shih, C.-L. C. Chien, Y.-H. Hsiao, and L.-C. Chen, "An Overview of the Development of a GPU with Integrated HBM on Silicon Interposer," *Electronic Components and Technology Conference*, pp. 1439–1444, 2016.

[57] J. Y. Xie, H. Shi, Y. Li, Z. Li, A. Rahman, K. Chandrasekar, D. Ratakonda, M. Deo, K. Chanda, and V. Hool, "Enabling the 2.5 D integration," in *International Symposium on Microelectronics*, vol. 2012, no. 1, pp. 000 254–000 267, 2012.

[58] Z. Li, H. Shi, J. Xie, and A. Rahman, "Development of an optimized power delivery system for 3D IC integration with TSV silicon interposer," *Electronic Components and Technology Conference*, pp. 678–682, 2012.

[59] *Xilinx Multi-node Technology Leadership Continues with UltraScale+ Portfolio "3D on 3D" Solutions*, https://www.xilinx.com/support/documentation/white_papers/wp472-3D-on-3D.pdf.

[60] P. Batude, M. Vinet, A. Pouydebasque, C. Le Royer, B. Previtali, C. Tabone, J.-M. Hartmann, L. Sanchez, L. Baud, and V. Carron, "Advances in 3D CMOS sequential integration," *Proceedings of IEEE International Electron Devices Meeting*, pp. 1–4, 2009.

[61] C. S. Tan, R. J. Gutmann, and L. R. Reif, *Wafer level 3-D ICs process technology.* Springer Science & Business Media, 2009.

[62] *International Technology Roadmap for Semiconductors, Interconnect, 2009*, http://www.itrs.net/.

[63] C. Laviron, B. Dunne, V. Lapras, P. Galbiati, D. Henry, F. Toia, S. Moreau, R. Anciant, C. Brunet-Manquat, and N. Sillon, “Via first approach optimisation for through silicon via applications,” *Electronic Components and Technology Conference*, pp. 14–19, 2009.

[64] A. Agarwal, R. B. Murthy, V. Lee, and G. Viswanadam, “Polysilicon interconnections (FEOL): Fabrication and characterization,” *Proceedings of Electronics Packaging Technology Conference*, pp. 317–320, 2009.

[65] P. Garrou, C. Bower, and P. Ramm, *Handbook of 3d integration: volume 1-technology and applications of 3D integrated circuits.* John Wiley & Sons, 2011.

[66] T. M. Bauer, S. L. Shinde, J. E. Massad, and D. L. Hetherington, “Front end of line integration of high density, electrically isolated, metallized through silicon vias,” *Electronic Components and Technology Conference*, pp. 1165–1169, 2009.

[67] G. Pares, S. Minoret, J. F. Lugand, S. Huet, V. Lapras, R. Anciant, D. Henry, N. Sillon, and B. Dunne, “Mid-process through silicon vias technology using tungsten metallization: Process optimazation and electrical results,” *Proceedings of Electronics Packaging Technology Conference*, pp. 772–777, 2009.

[68] W. L. Loh, Q. Ren, and K. J. Chui, “TSV via last etch integration challenges and etch optimization,” *Proceedings of Electronics Packaging Technology Conference*, pp. 1–3, 2015.

[69] G. Hwang and R. Kalaiselvan, “Development of TSV electroplating process for via-last technology,” *Electronic Components and Technology Conference*, pp. 67–72, 2017.

[70] S. Wang and M. B. Tahoori, “Electromigration-Aware Local-Via Allocation in Power/Ground TSVs of 3-D ICs,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 10, pp. 2881–2892, 2017.

[71] M. Puech, J.-M. Thevenoud, J. M. Gruffat, N. Launay, N. Arnal, and P. Godinat, “Fabrication of 3D packaging TSV using DRIE,” *Proceedings of IEEE International Symposium on Design, Test, Integration and Packaging of MEMS/MOEMS*, pp. 109–114, 2008.

[72] G. Pares, N. Bresson, S. Minoret, V. Lapras, P. Brianceau, J. F. Lugand, R. Anciant, and N. Sillon, “Through silicon via technology using tungsten metallization,” *Proceedings of International Conference on IC Design & Technology*, pp. 1–4, 2011.

[73] K. W. Lee, H. Hashimoto, M. Onishi, Y. Sato, M. Murugesan, J. C. Bea, T. Fukushima, T. Tanaka, and M. Koyanagi, “A resilient 3-D stacked multicore processor fabricated using die-level 3-D integration and backside TSV technologies,” *Electronic Components and Technology Conference*, pp. 304–308, 2014.

[74] R. Agarwal, D. Hiner, S. Kannan, K. Lee, D. Kim, J. Paek, S. Kang, Y. Song, S. Dej, and D. Smith, “TSV integration on 20nm logic Si: 3D assembly and reliability results,” *Electronic Components and Technology Conference*, pp. 590–595, 2014.

[75] D. J. Na, K. O. Aung, W. K. Choi, T. Kida, T. Ochiai, T. Hashimoto, M. Kimura, K. Kata, S. W. Yoon, and A. C. B. Yong, “TSV MEOL (Mid End of Line) and packaging technology of mobile 3D-IC stacking,” *Electronic Components and Technology Conference*, pp. 596–600, 2014.

[76] M.-J. Tsai, “Overview of ITRI’s TSV technology,” in *7th Annual SEMATECH Symposium Japan, June*, 2011, http://www.sematech.org/meetings/archives/symposia/9237/Session%205%203D%20interconnect/1%20MJ_Tsai_ITRI.pdf.

[77] H. Ikeda, “Heterogeneous 3D stacking technology developments in ASET,” in *CPMT Symposium Japan, 2012 2nd IEEE*, pp. 1–4, 2012.

[78] H. B. Chang, H. Y. Chen, P. C. Kuo, C. H. Chien, E. B. Liao, T. C. Lin, T. S. Wei, Y. C. Lin, Y. H. Chen, and K. F. Yang, "High-aspect ratio through silicon via (TSV) technology," *Proceedings of IEEE International Symposium on VLSI Technology*, pp. 173–174, 2012.

[79] E. Beyne, "Reliable via-middle copper through-silicon via technology for 3-D integration," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 6, no. 7, pp. 983–992, 2016.

[80] J. M. E. Harper, C. Cabral Jr, P. C. Andricacos, L. Gignac, I. C. Noyan, K. P. Rodbell, and C. K. Hu, "Mechanisms for microstructure evolution in electroplated copper thin films near room temperature," *Journal of Applied Physics*, vol. 86, no. 5, pp. 2516–2525, 1999.

[81] L. W. Kong, J. R. Lloyd, K. B. Yeap, E. Zschech, A. Rudack, M. Liehr, and A. Diebold, "Applying x-ray microscopy and finite element modeling to identify the mechanism of stress-assisted void growth in through-silicon vias," *Journal of Applied Physics*, vol. 110, no. 5, p. 053502, 2011.

[82] C. Okoro, L. E. Levine, R. Xu, K. Hummler, and Y. Obeng, "X-ray micro-beam diffraction measurement of the effect of thermal cycling on stress in Cu TSV: a comparative study," *Electronic Components and Technology Conference*, pp. 1648–1651, 2014.

[83] C. Okoro, R. Labie, K. Vanstreels, A. Franquet, M. Gonzalez, B. Vandevelde, E. Beyne, D. Vandepitte, and B. Verlinden, "Impact of the electrodeposition chemistry used for TSV filling on the microstructural and thermo-mechanical response of Cu," *Journal of Materials Science*, vol. 46, no. 11, pp. 3868–3882, 2011.

[84] I. De Wolf, V. Simons, V. Cherman, R. Labie, B. Vandevelde, and E. Beyne, "In-depth Raman spectroscopy analysis of various parameters affecting the mechanical stress near the surface and bulk of Cu-TSVs," *Electronic Components and Technology Conference*, pp. 331–337, 2012.

[85] S.-K. Ryu, K.-H. Lu, T. Jiang, J.-H. Im, R. Huang, and P. S. Ho, "Effect of thermal stresses on carrier mobility and keep-out zone around through-silicon vias for 3-D integration," *IEEE Transactions on Device and Materials Reliability*, vol. 12, no. 2, pp. 255–262, 2012.

[86] Y. Oba, J. De Messemaeker, A. M. Tyrovouzi, Y. Miyamori, J. De Vos, T. Wang, G. Beyer, E. Beyne, I. De Wolf, and K. Croes, "Effect of test structure on electromigration characteristics in three-dimensional through silicon via stacked devices," *Japanese Journal of Applied Physics*, vol. 54, no. 5S, p. 05EE01, 2015.

[87] T. Frank, S. Moreau, C. Chappaz, L. Arnaud, P. Leduc, A. Thuaire, and L. Anghel, "Electromigration behavior of 3D-IC TSV interconnects," *Electronic Components and Technology Conference*, pp. 326–330, 2012.

[88] T. Frank, S. Moreau, C. Chappaz, P. Leduc, L. Arnaud, A. Thuaire, E. Chery, F. Lorut, L. Anghel, and G. Poupon, "Reliability of TSV interconnects: Electromigration, thermal cycling, and impact on above metal level dielectric," *Microelectronics Reliability*, vol. 53, no. 1, pp. 17–29, 2013.

[89] C.-K. Hu, L. Gignac, E. Liniger, B. Herbst, D. L. Rath, S. T. Chen, S. Kaldor, A. Simon, and W.-T. Tseng, "Comparison of Cu electromigration lifetime in Cu interconnects coated with various caps," *Applied Physics Letters*, vol. 83, no. 5, pp. 869–871, 2003.

[90] N. H. Khan, S. M. Alam, and S. Hassoun, "System-level comparison of power delivery design for 2D and 3D ICs," *Proceedings of IEEE International Conference on 3D System Integration*, pp. 1–7, 2009.

[91] S. Wang, F. Firouzi, F. Oboril, and M. B. Tahoori, "P/G TSV planning for IR-drop reduction in 3D-ics," *Proceedings of Design, Automation, and Test in Europe*, pp. 480–485, 2014.

[92] T. Dao, D. H. Triyoso, M. Petras, and M. Canonico, “Through silicon via stress characterization,” *Proceedings of International Conference on IC Design & Technology*, pp. 1–3, 2009.

[93] J. S. Yang, K. Athikulwongse, Y. J. Lee, S. K. Lim, and D. Z. Pan, “TSV stress aware timing analysis with applications to 3D-IC layout optimization,” *Proceedings of ACM/IEEE Design Automation Conference*, pp. 803–806, 2010.

[94] K. H. Lu, S. K. Ryu, J. Im, R. Huang, and P. S. Ho, “Thermomechanical reliability of through-silicon vias in 3D interconnects,” *Proceedings of IEEE International Reliability Physics Symposium*, pp. 3D–1, 2011.

[95] M. Jung and S. K. Lim, “A study of IR-drop noise issues in 3D ICs with through-silicon-vias,” *Proceedings of IEEE International Conference on 3D System Integration*, pp. 1–7, 2010.

[96] M. Anis, M. Mahmoud, M. Elmasry, and S. Areibi, “Dynamic and leakage power reduction in MTCMOS circuits using an automated efficient gate clustering technique,” *Proceedings of ACM/IEEE Design Automation Conference*, pp. 480–485, 2002.

[97] M. B. Healy and S. K. Lim, “Power delivery system architecture for many-tier 3D systems,” *Electronic Components and Technology Conference*, pp. 1682–1688, 2010.

[98] G. Huang, M. Bakir, A. Naeemi, H. Chen, and J. D. Meindl, “Power delivery for 3D chip stacks: Physical modeling and design implication,” *Proceedings of Electrical Performance of Electronic Packaging*, pp. 205–208, 2007.

[99] H. Yu, J. Ho, and L. He, “Simultaneous power and thermal integrity driven via stapling in 3D ICs,” *Proceedings of International Conference on Computer-Aided Design*, pp. 802–808, 2006.

[100] S. K. Marella, S. V. Kumar, and S. S. Sapatnekar, “A holistic analysis of circuit timing variations in 3D-ICs with thermal and TSV-induced stress considerations,” *Proceedings of International Conference on Computer-Aided Design*, pp. 317–324, 2012.

[101] K. Athikulwongse, A. Chakraborty, J. S. Yang, D. Z. Pan, and S. K. Lim, “Stress-driven 3D-IC placement with TSV keep-out zone and regularity study,” *Proceedings of International Conference on Computer-Aided Design*, pp. 669–674, 2010.

[102] C. W. Liu and Y. W. Chang, “Floorplan and power/ground network co-synthesis for fast design convergence,” *Proceedings of ACM International Symposium on Physical Design*, pp. 86–93, 2006.

[103] P. Falkenstern, Y. Xie, Y. W. Chang, and Y. Wang, “Three-dimensional integrated circuits (3D IC) floorplan and power/ground network co-synthesis,” *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 169–174, 2010.

[104] Z. Li, Y. Ma, Q. Zhou, Y. Cai, Y. Wang, T. Huang, and Y. Xie, “Thermal-aware power network design for IR drop reduction in 3D ICs,” *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 47–52, 2012.

[105] S. Yao, X. Chen, Y. Wang, Y. Ma, Y. Xie, and H. Yang, “Efficient region-aware P/G TSV planning for 3D ICs,” *Proceedings of International Symposium on Quality of Electronic Design*, pp. 171–178, 2014.

[106] C. Jang, J. Kim, B. Ahn, and J. Chong, “Power bumps and through-silicon-vias placement with optimised power mesh structure for power delivery network in three-dimensional-integrated circuits,” *IET Computers & Digital Techniques*, vol. 7, no. 1, pp. 11–20, 2013.

[107] G. Katti, M. Stucchi, K. De Meyer, and W. Dehaene, "Electrical modeling and characterization of through silicon via for three-dimensional ICs," *IEEE Transactions on Electron Devices*, vol. 57, no. 1, pp. 256–262, 2010.

[108] N. H. Khan, S. M. Alam, and S. Hassoun, "Power delivery design for 3-D ICs using different through-silicon via (TSV) technologies," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 4, pp. 647–658, 2011.

[109] S. Wang, J. Chen, and M. Tehranipoor, "Representative critical reliability paths for low-cost and accurate on-chip aging evaluation," *Proceedings of International Conference on Computer-Aided Design*, pp. 736–741, 2012.

[110] F. Firouzi, F. Ye, K. Chakrabarty, and M. B. Tahoori, "Representative critical-path selection for aging-induced delay monitoring," *Proceedings of IEEE International Test Conference*, pp. 1–10, 2013.

[111] G. P. McCormick, "Computability of global solutions to factorable nonconvex programs: Part i - convex underestimating problems," *Mathematical programming*, vol. 10, no. 1, pp. 147–175, 1976.

[112] M. C. Tsai, T. C. Wang, and T. T. Hwang, "Through-silicon via planning in 3-D floorplanning," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 8, pp. 1448–1457, 2011.

[113] Q. B. Wang, J. Lillis, and S. Sanyal, "An LP-based methodology for improved timing-driven placement," *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 1139–1143, 2005.

[114] C. Albrecht, "IWLS 2005 benchmarks," 2005.

[115] 45nm nangate library, http://www.si2.org/openeda.si2.org/projects/nangatelib, December 2010.

[116] E. Beyne, P. De Moor, W. Ruythooren, R. Labie, A. Jourdain, H. Tilmans, D. S. Tezcan, P. Soussan, B. Swinnen, and R. Cartuyvels, "Through-silicon via and die stacking technologies for microsystems-integration," *Proceedings of IEEE International Electron Devices Meeting*, pp. 1–4, 2008.

[117] N. Serin, T. Serin, S. Horzum, and Y. Celik, "Annealing effects on the properties of copper oxide thin films prepared by chemical deposition," *Semiconductor science and technology*, vol. 20, no. 5, p. 398, 2005.

[118] J. Cong and G. Luo, "A 3D physical design flow based on open access," *Proceedings of International Conference on Communications, Circuits and Systems*, pp. 1103–1107, 2009.

[119] *IBM ILOG CPLEX optimizer*, http://www.ibm.com/.

[120] J. Pak, M. Pathak, S. K. Lim, and D. Z. Pan, "Modeling of electromigration in through-silicon-via based 3D IC," *Electronic Components and Technology Conference*, pp. 1420–1427, 2011.

[121] X. Zhao, Y. Wan, M. Scheuermann, and S. K. Lim, "Transient modeling of TSV-wire electromigration and lifetime analysis of power distribution network for 3D ICs," *Proceedings of International Conference on Computer-Aided Design*, pp. 363–370, 2013.

[122] B. Li, T. D. Sullivan, T. C. Lee, and D. Badami, "Reliability challenges for copper interconnects," *Microelectronics Reliability*, vol. 44, no. 3, pp. 365–380, 2004.

[123] B. Li, E. Yashchin, C. Christiansen, J. Gill, R. Filippi, and T. Sullivan, "Application of three-parameter lognormal distribution in EM data analysis," *Microelectronics Reliability*, vol. 46, no. 12, pp. 2049–2055, 2006.

[124] R. L. De Orio, H. Ceric, and S. Selberherr, “Electromigration failure in a copper dual-damascene structure with a through silicon via,” *Microelectronics Reliability*, vol. 52, no. 9, pp. 1981–1986, 2012.

[125] T. Frank, C. Chappaz, P. Leduc, L. Arnaud, S. Moreau, A. Thuaire, R. El Farhane, F. Lorut, and L. Anghel, “Resistance increase due to electromigration induced depletion under TSV,” *Proceedings of IEEE International Reliability Physics Symposium*, pp. 3F–4, 2011.

[126] J. Pak, S. K. Lim, and D. Z. Pan, “Electromigration study for multiscale power/ground vias in TSV-based 3D ICs,” *IEEE Transactions on Computer-Aided Design*, vol. 33, no. 12, pp. 1873–1885, 2014.

[127] V. Sukharev, A. Kteyan, and E. Zschech, “Physics-based models for EM and SM simulation in three-dimensional IC structures,” *IEEE Transactions on Device and Materials Reliability*, vol. 12, no. 2, pp. 272–284, 2012.

[128] T. Gupta, *Copper interconnect technology.* Springer Science & Business Media, 2010.

[129] M. Lin, N. Jou, J. W. Liang, and K. C. Su, “Effect of multiple via layout on electromigration performance and current density distribution in copper interconnect,” *Proceedings of IEEE International Reliability Physics Symposium*, pp. 844–847, 2009.

[130] G. Xu, L.-D. Huang, D. Z. Pan, and M. D. F. Wong, “Redundant-via enhanced maze routing for yield improvement,” *Proceedings of IEEE Asia South Pacific Design Automation Conference*, vol. 2, pp. 1148–1151, 2005.

[131] H.-Y. Chen, M.-F. Chiang, Y.-W. Chang, L. Chen, and B. Han, “Full-chip routing considering double-via insertion,” *IEEE Transactions on Computer-Aided Design*, vol. 27, no. 5, pp. 844–857, 2008.

[132] K.-Y. Lee and T.-C. Wang, “Post-routing redundant via insertion for yield/reliability improvement,” *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 303–308, 2006.

[133] S.-T. Lin, K.-Y. Lee, T.-C. Wang, C.-K. Koh, and K.-Y. Chao, “Simultaneous redundant via insertion and line end extension for yield optimization,” *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 633–638, 2011.

[134] J. Pak, B. Yu, and D. Z. Pan, “Electromigration-aware redundant via insertion,” *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 544–549, 2015.

[135] A. Todri-Sanial, S. Kundu, P. Girard, A. Bosio, L. Dilillo, and A. Virazel, “Globally constrained locally optimized 3D power delivery networks,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 10, pp. 2131–2144, 2014.

[136] Y. Peng, B. W. Ku, Y. Park, K.-I. Park, S.-J. Jang, J. S. Choi, and S. K. Lim, “Design, packaging, and architectural policy co-optimization for DC power integrity in 3D DRAM,” *Proceedings of ACM/IEEE Design Automation Conference*, pp. 1–6, 2015.

[137] C. Li, M. Xie, C.-K. Koh, J. Cong, and P. H. Madden, “Routability-driven placement and white space allocation,” *Proceedings of International Conference on Computer-Aided Design*, pp. 394–401, 2004.

[138] P. Saxena, R. S. Shelar, and S. Sapatnekar, *Routing Congestion in VLSI Circuits: Estimation and Optimization.* Springer Science & Business Media, 2006.

[139] R. Kastner, E. Bozorgzadeh, and M. Sarrafzadeh, “Pattern routing: use and theory for increasing predictability and avoiding coupling,” *IEEE Transactions on Computer-Aided Design*, vol. 21, no. 7, pp. 777–790, 2002.

[140] C.-C. Chang, J. Cong, Z. Pan, and X. Yuan, “Multilevel global placement with congestion control,” *IEEE Transactions on Computer-Aided Design*, vol. 22, no. 4, pp. 395–409, 2003.

[141] S. Chatterjee, M. Fawaz, and F. N. Najm, “Redundancy-aware electromigration checking for mesh power grids,” *Proceedings of International Conference on Computer-Aided Design*, pp. 540–547, 2013.

[142] T. Kim, B. Zheng, H.-B. Chen, Q. Zhu, V. Sukharev, and S. X.-D. Tan, “Lifetime optimization for real-time embedded systems considering electromigration effects,” *Proceedings of International Conference on Computer-Aided Design*, pp. 434–439, 2014.

[143] D.-A. Li and M. Marek-Sadowska, “Estimating true worst currents for power grid electromigration analysis,” *Proceedings of International Symposium on Quality of Electronic Design*, pp. 708–714, 2014.

[144] D.-A. Li, M. Marek-Sadowska, and S. R. Nassif, “A method for improving power grid resilience to electromigration-caused via failures,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 1, pp. 118–130, 2015.

[145] S. Wang, F. Firouzi, F. Oboril, and M. B. Tahoori, “Stress-aware P/G TSV planning in 3D-ICs,” *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 94–99, 2015.

[146] X. Zhao and S. K. Lim, “Through-silicon-via-induced obstacle-aware clock tree synthesis for 3D ICs,” *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 347–352, 2012.

[147] J. Knechtel, I. L. Markov, J. Lienig, and M. Thiele, “Multiobjective optimization of deadspace, a critical resource for 3D-IC integration,” *Proceedings of International Conference on Computer-Aided Design*, pp. 705–712, 2012.

[148] Opencores benchmark, http://opencores.org.

[149] J. Meng, K. Kawakami, and A. K. Coskun, “Optimizing energy efficiency of 3-D multicore systems with stacked DRAM under power and thermal constraints,” *Proceedings of ACM/IEEE Design Automation Conference*, pp. 648–655, 2012.

[150] F. Firouzi, S. Kiamehr, M. Tahoori, and S. Nassif, “Incorporating the impacts of workload-dependent runtime variations into timing analysis,” *Proceedings of Design, Automation, and Test in Europe*, pp. 1022–1025, 2013.

[151] I. Loi, S. Mitra, T. H. Lee, S. Fujita, and L. Benini, “A low-overhead fault tolerance scheme for TSV-based 3D network on chip links,” *Proceedings of International Conference on Computer-Aided Design*, pp. 598–602, 2008.

[152] B. Swinnen, W. Ruythooren, P. De Moor, L. Bogaerts, L. Carbonell, K. De Munck, B. Eyckens, S. Stoukatch, D. Sabuncuoglu Tezcan, Z. Tokei *et al.*, “3D integration by Cu-Cu thermo-compression bonding of extremely thinned bulk-Si die containing 10 μm pitch through-Si vias,” *Proceedings of IEEE International Electron Devices Meeting*, pp. 1–4, 2006.

[153] H. H. Lee and K. Chakrabarty, “Test challenges for 3D integrated circuits,” *IEEE Design & Test of Computers*, vol. 26, no. 5, pp. 26–35, 2009.

[154] L. Jiang, R. Ye, and Q. Xu, “Yield enhancement for 3D-stacked memory by redundancy sharing across dies,” *Proceedings of International Conference on Computer-Aided Design*, pp. 230–234, 2010.

[155] C. W. Chou, Y. J. Huang, and J. F. Li, “Yield-enhancement techniques for 3D random access memories,” *Proceedings of IEEE International Symposium on VLSI Design Automation and Test*, pp. 104–107, 2010.

[156] C. Ferri, S. Reda, and R. I. Bahar, “Strategies for improving the parametric yield and profits of 3D ICs,” *Proceedings of International Conference on Computer-Aided Design*, pp. 220–226, 2007.

[157] A. C. Hsieh and T. T. Hwang, “TSV redundancy: architecture and design issues in 3-D IC,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 4, pp. 711–722, 2012.

[158] Y. Zhao, S. Khursheed, and B. M. Al-Hashimi, “Cost-effective TSV grouping for yield improvement of 3D-ICs,” *Proceedings of IEEE Asian Test Symposium*, pp. 201–206, 2011.

[159] J. Xie, Y. Wang, and Y. Xie, “Yield-aware time-efficient testing and self-fixing design for TSV-based 3D ICs,” *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 738–743, 2012.

[160] L. Jiang, Q. Xu, and B. Eklow, “On effective TSV repair for 3D-stacked ICs,” *Proceedings of Design, Automation, and Test in Europe*, pp. 793–798, 2012.

[161] F. Ye and K. Chakrabarty, “TSV open defects in 3D integrated circuits: Characterization, test, and optimal spare allocation,” *Proceedings of ACM/IEEE Design Automation Conference*, pp. 1024–1030, 2012.

[162] Y. G. Chen, K. Y. Lai, M. C. Lee, Y. Shi, W. K. Hon, and S. C. Chang, “Yield and timing constrained spare TSV assignment for three-dimensional integrated circuits,” *Proceedings of Design, Automation, and Test in Europe*, p. 105, 2014.

[163] W. H. Lo, K. Chi, and T. T. Hwang, “Architecture of ring-based redundant TSV for clustered faults,” *Proceedings of Design, Automation, and Test in Europe*, pp. 848–853, 2015.

[164] H. Yan, Z. Li, Q. Zhou, and X. Hong, “Via assignment algorithm for hierarchical 3D placement,” *Proceedings of International Conference on Communications, Circuits and Systems*, vol. 2, 2005.

[165] D. H. Kim, K. Athikulwongse, and S. K. Lim, “A study of through-silicon-via impact on the 3D stacked IC layout,” *Proceedings of International Conference on Computer-Aided Design*, pp. 674–680, 2009.

[166] E. Singh, “Impact of radial defect clustering on 3D stacked IC yield from wafer to wafer stacking,” *Proceedings of IEEE International Test Conference*, pp. 1–7, 2012.

[167] E. Singh, “Analytical modeling of 3D stacked IC yield from wafer to wafer stacking with radial defect clustering,” *Proceedings of IEEE International Conference on Embedded Systems & VLSI Design*, pp. 26–31, 2014.

[168] C. L. Lung, J. H. Chien, Y. Shi, and S. C. Chang, “TSV fault-tolerant mechanisms with application to 3D clock networks,” *Proceedings of IEEE International Conference on SOC Design*, pp. 127–130, 2011.

[169] Y.-J. Huang and J.-F. Li, “Built-in self-repair scheme for the TSVs in 3-D ICs,” *IEEE Transactions on Computer-Aided Design*, vol. 31, no. 10, pp. 1600–1613, 2012.

[170] M. Jung, J. Mitra, D. Z. Pan, and S. K. Lim, “Tsv stress-aware full-chip mechanical reliability analysis and optimization for 3D IC,” *Proceedings of ACM/IEEE Design Automation Conference*, pp. 803–806, 2011.

[171] B. Kim, C. Sharbono, T. Ritzdorf, and D. Schmauch, “Factors affecting copper filling process within high aspect ratio deep vias for 3D chip stacking,” *Electronic Components and Technology Conference*, pp. 6–pp, 2006.

[172] A. P. Karmarkar, X. Xu, and V. Moroz, “Performanace and reliability analysis of 3D-integration structures employing through silicon via (TSV),” *Proceedings of IEEE International Reliability Physics Symposium*, pp. 682–687, 2009.

[173] L. Jiang, Q. Xu, and B. Eklow, “On effective through-silicon via repair for 3-D stacked ICs,” *IEEE Transactions on Computer-Aided Design*, vol. 32, no. 4, pp. 559–571, 2013.

[174] Y. G. Chen, W. Y. Wen, Y. Shi, W. K. Hon, and S. C. Chang, “Novel spare TSV deployment for 3D ICs considering yield and timing constraints,” *IEEE Transactions on Computer-Aided Design*, vol. 34, no. 4, pp. 577–588, 2014.

[175] Y. Zhao, “Investigation into yield and reliability enhancement of tsv-based three-dimensional integration circuits,” Ph.D. dissertation, University of Southampton, 2014.

[176] M. B. Tahoori, “Defects, yield, and design in sublithographic nano-electronics,” *Proceedings of IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 3–11, 2005.

[177] F. J. Meyer and D. K. Pradhan, “Modeling defect spatial distribution,” *IEEE Transactions on Computers*, vol. 38, no. 4, pp. 538–546, 1989.

[178] T. M. Breuel, “Two geometric algorithms for layout analysis,” *Proceedings of ACM International Workshop on Document Analysis Systems*, pp. 188–199, 2002.

[179] I. Koren and Z. Koren, “Defect tolerance in VLSI circuits: techniques and yield analysis,” *Proceedings of the IEEE*, vol. 86, no. 9, pp. 1819–1838, 1998.

[180] W. Kuo and M. J. Zuo, *Optimal reliability modeling: principles and applications*. John Wiley & Sons, 2003.

[181] A. M. Rushdi, “Utilization of symmetric switching functions in the computation of k-out-of-n system reliability,” *Microelectronics Reliability*, vol. 26, no. 5, pp. 973–987, 1986.

[182] J. Lillis, C.-K. Cheng, T.-T. Y. Lin, and C.-Y. Ho, “New performance driven routing techniques with explicit area/delay tradeoff and simultaneous wire sizing,” *Proceedings of ACM/IEEE Design Automation Conference*, pp. 395–400, 1996.

[183] F. Gao and J. P. Hayes, “Exact and heuristic approaches to input vector control for leakage power reduction,” *IEEE Transactions on Computer-Aided Design*, vol. 25, no. 11, pp. 2564–2571, 2006.

[184] R. Marculescu, D. Marculescu, and M. Pedram, “Sequence compaction for power estimation: Theory and practice,” *IEEE Transactions on Computer-Aided Design*, vol. 18, no. 7, pp. 973–993, 1999.

[185] S. Wang, M. B. Tahoori, and K. Chakrabarty, “Defect clustering-aware spare-TSV allocation for 3D ICs,” *Proceedings of International Conference on Computer-Aided Design*, pp. 307–314, 2015.

[186] Q. Xu, S. Chen, X. Xu, and B. Yu, “Clustered Fault Tolerance TSV Planning for 3D Integrated Circuits,” *IEEE Transactions on Computer-Aided Design*, vol. 36, no. 8, pp. 1287–1300, 2017.

[187] C. Serafy and A. Srivastava, "Online TSV health monitoring and built-in self-repair to overcome aging," *Proceedings of IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 224–229, 2013.

[188] X. Huang, V. Sukharev, T. Kim, H. Chen, and S. X.-D. Tan, "Electromigration recovery modeling and analysis under time-dependent current and temperature stressing," *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 244–249, 2016.

[189] X. Huang, V. Sukharev, T. Kim, and S. X.-D. Tan, "Dynamic electromigration modeling for transient stress evolution and recovery under time-dependent current and temperature stressing," *Integration, the VLSI journal*, 2016.

[190] K. D. Lee, "Electromigration recovery and short lead effect under bipolar-and unipolar-pulse current," *Proceedings of IEEE International Reliability Physics Symposium*, pp. 6B–3, 2012.

[191] M. H. Lin and A. S. Oates, "AC and pulsed-DC stress electromigration failure mechanisms in Cu interconnects," *Proceedings of IEEE International Interconnect Technology Conference*, pp. 1–3, 2013.

[192] J. Pak, S. K. Lim, and D. Z. Pan, "Electromigration-aware routing for 3D ICs with stress-aware EM modeling," *Proceedings of International Conference on Computer-Aided Design*, pp. 325–332, 2012.

[193] T. Kim, Z. Sun, C. Cook, J. Gaddipati, H. Wang, H. Chen, and S. X.-D. Tan, "Dynamic reliability management for near-threshold dark silicon processors," *Proceedings of International Conference on Computer-Aided Design*, pp. 1–7, 2016.

[194] M. Nicolaidis, V. Pasca, and L. Anghel, "Through-silicon-via built-in self-repair for aggressive 3D integration," *Proceedings of IEEE International Symposium on On-Line Testing*, pp. 91–96, 2012.

[195] J. L. Bonebakker, "Finding representative workloads for computer system design," 2007.

[196] W. Michiels, J. Korst, E. Aarts, and J. Van Leeuwen, "Performance ratios for the differencing method applied to the balanced number partitioning problem," *Annual Symposium on Theoretical Aspects of Computer Science*, pp. 583–595, 2003.

[197] J. Cong and Y. Zhang, "Thermal-driven multilevel routing for 3-D ICs," *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 121–126, 2005.

[198] M. Laisne, K. Arabi, and T. Petrov, "Systems and Methods Utilizing Redundancy in Semiconductor Chip Interconnects," *US Patent 8384417*, 2013.

[199] K. He, X. Huang, and X. D. Tan, "EM-based on-chip aging sensor for detection and prevention of counterfeit and recycled ICs," *Proceedings of International Conference on Computer-Aided Design*, pp. 146–151, 2015.

[200] Y. Xie, J. Cong, and S. S. Sapatnekar, *Three-dimensional integrated circuit design.* Springer-Verlag, 2010.

[201] G. Van der Plas, P. Limaye, I. Loi, A. Mercha, H. Oprins, C. Torregiani, S. Thijs, D. Linten, M. Stucchi, G. Katti *et al.*, "Design issues and considerations for low-cost 3-D TSV IC technology," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 293–307, 2011.

[202] B. Noia and K. Chakrabarty, "Pre-bond probing of TSVs in 3D stacked ICs," *Proceedings of IEEE International Test Conference*, pp. 1–10, 2011.

[203] S. Wang, Z. Sun, Y. Chen, S. X.-D. Tan, and M. B. Tahoori, "Leverage recovery effect to reduce electromigration degradation in power/ground TSV," *Proceedings of International Conference on Computer-Aided Design*, 2017.

[204] H.-B. Chen, S. X.-D. Tan, X. Huang, T. Kim, and V. Sukharev, "Analytical Modeling and Characterization of Electromigration Effects for Multibranch Interconnect Trees," *IEEE Transactions on Computer-Aided Design*, vol. 35, no. 11, pp. 1811–1824, 2016.

[205] J.-H. Choy, V. Sukharev, S. Chatterjee, F. N. Najm, A. Kteyan, and S. Moreau, "Finite-difference methodology for full-chip electromigration analysis applied to 3D IC test structure: Simulation vs. experiment," *Proceedings of International Conference on Simulation of Semiconductor Processes and Devices*, pp. 41–44, 2017.

[206] M. Jackson, "A silicon interposer-based 2.5 D-IC design flow, going 3D by evolution rather than by revolution," *Synopsis Insight Newsletter, Tech. Rep*, no. 1, 2012.

[207] M. Sunohara, T. Tokunaga, T. Kurihara, and M. Higashi, "Silicon interposer with TSVs (Through Silicon Vias) and fine multilayer wiring," *Electronic Components and Technology Conference*, pp. 847–852, 2008.

[208] K. Kumagai, Y. Yoneda, H. Izumino, H. Shimojo, M. Sunohara, T. Kurihara, M. Higashi, and Y. Mabuchi, "A silicon interposer BGA package with Cu-filled TSV and multi-layer Cu-plating interconnect," *Electronic Components and Technology Conference*, pp. 571–576, 2008.

[209] B. Banijamali, S. Ramalingam, K. Nagarajan, and R. Chaware, "Advanced reliability study of TSV interposers and interconnects for the 28nm technology FPGA," *Electronic Components and Technology Conference*, pp. 285–290, 2011.

[210] *International Technology Roadmap for Semiconductors, Heterogeneous Integration, 2015*, ITRS, 2015, http://www.itrs.net/.

[211] R. Wang, K. Chakrabarty, and S. Bhawmik, "Interconnect testing and test-path scheduling for interposer-based 2.5-D ICs," *IEEE Transactions on Computer-Aided Design*, vol. 34, no. 1, pp. 136–149, 2015.

[212] R. Wang, G. Li, R. Li, J. Qiang, and K. Chakrabarty, "Extest scheduling and optimization for 2.5D SoCs with wrapped tiles," *IEEE Transactions on Computer-Aided Design*, vol. 36, no. 6, pp. 1030–1042, 2017.

[213] J. Rearick, "Testing the AMD FIJI GPU in the 3rd Dimension," *Keynote Speech, ITC 3D Test Workshop*, 2015.

[214] R. Wang, K. Chakrabarty, and S. Bhawmik, "Built-in self-test and test scheduling for interposer-based 2.5D IC," *ACM Transactions on Design Automation of Electronic Systems*, vol. 20, no. 4, p. 58, 2015.

[215] K. Chakrabarty, "Test scheduling for core-based systems using mixed-integer linear programming," *IEEE Transactions on Computer-Aided Design*, vol. 19, no. 10, pp. 1163–1174, 2000.

[216] Y. Huang, W.-T. Cheng, C.-C. Tsai, N. Mukherjee, O. Samman, Y. Zaidan, and S. M. Reddy, "Resource allocation and test scheduling for concurrent test of core-based SOC design," *Proceedings of IEEE Asian Test Symposium*, pp. 265–270, 2001.

[217] V. Iyengar and K. Chakrabarty, "System-on-a-chip test scheduling with precedence relationships, preemption, and power constraints," *IEEE Transactions on Computer-Aided Design*, vol. 21, no. 9, pp. 1088–1094, 2002.

[218] S. Koranne, "On test scheduling for core-based SOCs," *Proceedings of IEEE Asia South Pacific Design Automation Conference*, p. 505, 2002.

[219] W. Zou, S. M. Reddy, I. Pomeranz, and Y. Huang, "SOC test scheduling using simulated annealing," *Proceedings IEEE VLSI Test Symposium*, pp. 325–330, 2003.

[220] J. Pouget, E. Larsson, and Z. Peng, "SOC test time minimization under multiple constraints." *Proceedings of IEEE Asian Test Symposium*, pp. 312–317, 2003.

[221] E. Larsson, K. Arvidsson, H. Fujiwara, and Z. Peng, "Efficient test solutions for core-based designs," *IEEE Transactions on Computer-Aided Design*, vol. 23, no. 5, pp. 758–775, 2004.

[222] Q. Xu and N. Nicolici, "Resource-constrained system-on-a-chip test: a survey," *IEE Proceedings, Computers and Digital Techniques*, vol. 152, no. 1, pp. 67–81, 2005.

[223] Z. G and H. Ito, "Concurrent core test for SOC using shared test set and scan chain disable," *Proceedings of Design, Automation, and Test in Europe*, pp. 1045–1050, 2006.

[224] Q. Zhou and K. J. Balakrishnan, "Test cost reduction for SoC using a combined approach to test data compression and test scheduling," *Proceedings of Design, Automation, and Test in Europe*, pp. 39–44, 2007.

[225] C. C. Chi, E. J. Marinissen, S. K. Goel, and C.-W. Wu, "Post-bond testing of 2.5D-SICs and 3D-SICs containing a passive silicon interposer base," *Proceedings of IEEE International Test Conference*, pp. 1–10, 2011.

[226] C.-C. Chi, E. J. Marinissen, S. K. Goel, and C.-W. Wu, "Multi-visit TAMs to reduce the post-bond test length of 2.5D-SICs with a passive silicon interposer base," *Proceedings of IEEE Asian Test Symposium*, pp. 451–456, 2011.

[227] Y. Deng and W. P. Maly, "2.5-dimensional VLSI system integration," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 6, pp. 668–677, 2005.

[228] M. Bohr, "The new era of scaling in an SoC world," *Proceedings of International Solid State Circuits Conference*, pp. 23–28, 2009.

[229] K. Miyase, K. Noda, H. Ito, K. Hatayama, T. Aikyo, Y. Yamato, H. Furukawa, X. Wen, and S. Kajihara, "Effective IR-drop reduction in at-speed scan testing using distribution-controlling X-identification," *Proceedings of International Conference on Computer-Aided Design*, pp. 52–58, 2008.

[230] X. Zhang, J. K. Lin, S. Wickramanayaka, S. Zhang, R. Weerasekera, R. Dutta, K. F. Chang, K.-J. Chui, H. Y. Li, and D. S. Wee H., "Heterogeneous 2.5D integration on through silicon interposer," *Applied Physics Reviews*, vol. 2, no. 2, p. 021308, 2015.

[231] E. J. Marinissen, "Challenges and emerging solutions in testing TSV-based 2 1/2D-and 3D-stacked ICs," *Proceedings of Design, Automation, and Test in Europe*, pp. 1277–1282, 2012.

[232] S. Y. Huang, L. R. Huang, K. H. Tsai, and W. T. Cheng, "Delay testing and characterization of post-bond interposer wires in 2.5-D ICs," *Proceedings of IEEE International Test Conference*, pp. 1–8, 2013.

[233] R. Wang, K. Chakrabarty, and B. Eklow, "Scan-based testing of post-bond silicon interposer interconnects in 2.5-D ICs," *IEEE Transactions on Computer-Aided Design*, vol. 33, no. 9, pp. 1410–1423, 2014.

[234] R. Wang, B. Bhaskaran, K. Natarajan, A. Abdollahian, K. Narayanun, K. Chakrabarty, and A. Sanghani, “A programmable method for low-power scan shift in SoC integrated circuits,” *Proceedings IEEE VLSI Test Symposium*, pp. 1–6, 2016.

[235] H. Jun, S. Nam, H. Jin, J. C. Lee, Y. J. Park, and J. J. Lee, “High-bandwidth memory (HBM) test challenges and solutions,” *IEEE Design & Test of Computers*, vol. 34, no. 1, pp. 16–25, 2017.

[236] H. H. Jones, Technical Viability of Stacked Silicon Interconnect Technology. Xilinx. White Paper, http://www.xilinx.com/publications/technology/stacked-siliconinterconnect-technology-ibs-research.pdf, October 2010.

[237] P. Dorsey, Xilinx Stacked Silicon Interconnect Technology Delivers Breakthrough FPGA Capacity, Bandwidth, and Power Efficiency. White Paper, http://www.xilinx.com/support/documentation/whitepapers/wp380StackedSiliconInterconnectTechnology.pdf, October 2010.

[238] R. Dorsch, R. H. Rivera, H. J. Wunderlich, and M. Fischer, “Adapting an SoC to ATE concurrent test capabilities,” *Proceedings of IEEE International Test Conference*, pp. 1169–1175, 2002.

[239] K. J. Lee, J. J. Chen, and C. H. Huang, “Using a single input to support multiple scan chains,” *Proceedings of International Conference on Computer-Aided Design*, pp. 74–78, 1998.

[240] J. H. Jiang, W.-B. Jone, S.-C. Chang, and S. Ghosh, “Embedded core test generation using broadcast test architecture and netlist scrambling,” *IEEE Transactions on Reliability*, vol. 52, no. 4, pp. 435–443, 2003.

[241] G. John, Test Flow for Advanced Packages (2.5 D/SLIM/3D). White Paper, http://www.circuitnet.com/articles/111078.html, December 2015.

[242] Y. Liu, M. Li, M. Jiang, D. W. Kim, S. Gu, and K. N. Tu, “Joule Heating Enhanced Electromigration Failure in Redistribution Layer in 2.5D IC,” *Electronic Components and Technology Conference*, pp. 1359–1363, 2016.

[243] *IEEE Std 1149.1^{TM}-2001, IEEE Standard Test Access Port and Boundary-Scan Architecture*, IEEE Computer Society, IEEE, New York, NY, USA, 2001.

[244] E. J. Marinissen, T. McLaurin, and H. Jiao, “IEEE Std P1838: DfT standard-under-development for 2.5D-, 3D-, and 5.5D-SICs,” *Proceedings of IEEE European Test Symposium*, pp. 1–10, 2016.

[245] M. J. Wang, C. Y. Hung, C. L. Kao, P. N. Lee, C. H. Chen, C. P. Hung, and H. M. Tong, “TSV technology for 2.5D IC solution,” *Electronic Components and Technology Conference*, pp. 284–288, 2012.

[246] R. Wang, Z. Li, S. Kannan, and K. Chakrabarty, “Pre-bond testing of the silicon interposer in 2.5D ICs,” *Proceedings of Design, Automation, and Test in Europe*, pp. 978–983, 2016.

[247] A. Larsson, “Test Optimization for Core-based System-on-Chip,” Ph.D. dissertation, Linkoping University, 2008.

[248] *TurboBIST-Logic^{TM} Logic BIST Tool Suite*, SynTest, http://www.syntest.com/ProdDataSheet/TBISTLOGIC-datasheet081502a.pdf, 2008.

[249] P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, and H.-J. Wunderlich, “A modified clock scheme for a low power BIST test pattern generator,” *Proceedings IEEE VLSI Test Symposium*, pp. 306–311, 2001.

[250] X. Lin, R. Press, J. Rajski, P. Reuter, T. Rinderknecht, B. Swanson, and N. Tamarapalli, “High-frequency, at-speed scan testing,” *IEEE Design & Test of Computers*, vol. 20, no. 5, pp. 17–25, 2003.

[251] J. Nyathi, S. Sarkar, and P. P. Pande, "Multiple clock domain synchronization for network on chip architectures," *Proceedings of IEEE International Conference on SOC Design*, pp. 291–294, 2007.

[252] P. Wohl, J. A. Waicukauski, J. E. Colburn, and M. Sonawane, "Achieving extreme scan compression for SoC designs," *Proceedings of IEEE International Test Conference*, pp. 1–8, 2014.

[253] P. K. D. Jagannadha, M. Yilmaz, M. Sonawane, S. Chadalavada, S. Sarangi, B. Bhaskaran, and A. Abdollahian, "Advanced test methodology for complex SoCs," *Proceedings of IEEE International Test Conference*, pp. 1–10, 2016.

[254] C. Mead, "Neuromorphic electronic systems," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1629–1636, 1990.

[255] T. Duong, S. Kemeny, M. Tran, T. Daud, A. Thakoor, D. Ludwig, C. Saunders, and J. Carson, "Low power analog neurosynapse chips for a 3-D "sugarcube" neuroprocessor," in *Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on*, vol. 3, pp. 1907–1911, 1994.

[256] A. Bermak, "A highly scalable 3D chip for binary neural network classification applications," *Proceedings of International Symposium on Circuits and Systems*, vol. 5, pp. V685–V688, 2003.

[257] D. Kim, J. Kung, S. Chai, S. Yalamanchili, and S. Mukhopadhyay, "Neurocube: A programmable digital neuromorphic architecture with high-density 3D memory," *Proceedings of IEEE International Symposium on Computer Architecture*, pp. 380–392, 2016.

[258] F. Clermidy, R. Heliot, A. Valentian, C. Gamrat, O. Bichler, M. Duranton, B. Blehadj, and O. Temam, "Advanced technologies for brain-inspired computing," *Proceedings of IEEE Asia South Pacific Design Automation Conference*, pp. 563–569, 2014.

[259] B. Belhadj, A. Valentian, P. Vivet, M. Duranton, L. He, and O. Temam, "The improbable but highly appropriate marriage of 3D stacking and neuromorphic accelerators," *Proceedings of IEEE International Conference on Compilers, Architecture and Synthesis for Embedded Systems*, pp. 1–6, 2014.