ABSTRACT

Title of dissertation: ELECTRO-THERMAL CODESIGN IN LIQUID COOLED 3D ICS: PUSHING THE POWER-PERFORMANCE LIMITS

Bing Shi, Doctor of Philosophy, 2013

Dissertation directed by: Professor Ankur Srivastava
Department of Electrical and Computer Engineering

The performance improvement of today's computer systems is usually accompanied by increased chip power consumption and system temperature. Modern CPUs dissipate an average of $70-100W$ power while spatial and temporal power variations result in hotspots with even higher power density (up to $300W/cm^2$). The coming years will continue to witness a significant increase in CPU power dissipation due to advanced multi-core architectures and 3D integration technologies. Nowadays the problems of increased chip power density, leakage power and system temperatures have become major obstacles for further improvement in chip performance. The conventional air cooling based heat sink has been proved to be insufficient for three dimensional integrated circuits (3D-ICs). Hence better cooling solutions are necessary. Micro-fluidic cooling, which integrates micro-channel heat sinks into silicon substrates of the chip and uses liquid flow to remove heat inside the chip, is an effective active cooling scheme for 3D-ICs. While the micro-fluidic cooling provides excellent cooling to 3D-ICs, the associated overhead (cooling power

consumed by the pump to inject the coolant through micro-channels) is significant. Moreover, the 3D-IC structure also imposes constraints on micro-channel locations (basically resource conflict with through-silicon-vias TSVs or other structures).

In this work, we investigate optimized micro-channel configurations that address the aforementioned considerations. We develop three micro-channel structures (hotspot optimized cooling configuration, bended micro-channel and hybrid cooling network) that can provide sufficient cooling to 3D-IC with minimum cooling power overhead, while at the same time, compatible with the existing electrical structure such as TSVs. These configurations can achieve up to 70% cooling power savings compared with the configuration without any optimization. Based on these configurations, we then develop a micro-fluidic cooling based dynamic thermal management approach that maintains the chip temperature through controlling the fluid flow rate (pressure drop) through micro-channels. These cooling configurations are designed after the electrical parts, and therefore, compatible with the current standard IC design flow.

Furthermore, the electrical, thermal, cooling and mechanical aspects of 3D-IC are interdependent. Hence the conventional design flow that designs the cooling configuration after electrical aspect is finished will result in inefficiencies. In order to overcome this problem, we then investigate electrical-thermal co-design methodology for 3D-ICs. Two co-design problems are explored: TSV assignment and micro-channel placement co-design, and gate sizing and fluidic cooling co-design. The experimental results show that the co-design enables a fundamental power-performance improvement over the conventional design flow which separates the

electrical and cooling design. For example, the gate sizing and fluidic cooling co-design achieves 12% power savings under the same circuit timing constraint and 16% circuit speedup under the same power budget.

ELECTRO-THERMAL CODESIGN IN LIQUID COOLED 3D ICS:
PUSHING THE POWER-PERFORMANCE LIMITS

by

Bing Shi

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2013

Advisory Committee:
Professor Ankur Srivastava, Chair/Advisor
Professor Joseph JaJa
Professor Shuvra Bhattacharyya
Professor Donald Yeung
Professor Doron Levy

© Copyright by
Bing Shi
2013

ACKNOWLEDGEMENT

I would like to thank my advisor, Professor Ankur Srivastava for the support and guidance he has provided throughout my time in the Ph.D. program. Thank you for introducing me to the world of Electronic Design Automation, for giving me so many opportunities, for helping me every step of the way, for encouraging me in those hard times.

In addition, I would like to thank Professor Joseph JaJa who helped me a lot in my Ph.D. oral qualify exam, research proposal and also Ph.D. dissertation. I would like to thank Professor Shuvra Bhattacharyya for his support on my competition for ECE dissertation fellowship.

I would also like to thank my committee members, Professor Joseph JaJa, Professor Shuvra Bhattacharyya, Professor Donald Yeung and Professor Doron Levy, for their time, comments and feedback.

I also thank all past and present members of our lab: Domenic Forte, Yufu Zhang, Caleb Serafy, Tiantao Lu and Chongxi Bao, for their help, friendship, and support. I am grateful for all the fun times we have shared throughout the years.

Finally, I would like to thank my parents and my family for their ongoing support and encouragement. Thank you for all of their love and support over the course of my long journey as an academic.

Table of Contents

List of Figures vii

List of Tables ix

1 Introduction 1
1.1 Thermal Issues in 3D-ICs . . . 2
1.2 Conventional Dynamic Thermal Management . . . 4
1.3 Interlayer Micro-fluidic Cooling . . . 6
1.4 Interdependency between Electrical, Thermal, Reliability and Cooling 7
1.5 Advantage of Electrical and Cooling System Co-Design . . . 8
1.6 Thesis Outline . . . 9

2 Background 11
2.1 Basics of Three Dimensional Integrated Circuit . . . 11
2.2 Fundamental Characteristics of Fluids in Micro-channels . . . 12
2.2.1 Conservation Law of Fluid Dynamics . . . 12
2.2.2 Dimensionless Numbers in Fluid Mechanics . . . 13
2.2.3 Single and Two Phase Flow . . . 15
2.2.4 Laminar and Turbulent Flow . . . 17
2.3 Thermal Modeling of 3D-IC with Micro-fluidic Cooling . . . 19
2.3.1 Distributed RC Thermal Model . . . 19
2.3.2 Cooling Performance of Micro-channels . . . 20
2.3.3 Overall Thermal Model of 3D-IC with Micro-channels . . . 22
2.3.4 Thermal Impact of TSVs . . . 23
2.4 Modeling of Power Consumption . . . 24
2.4.1 Dynamic Power Consumption . . . 25
2.4.2 Leakage Power Consumption . . . 26
2.4.3 Micro-channel Cooling Power . . . 27
2.4.3.1 Straight Micro-channels . . . 27
2.4.3.2 Micro-channels with Bends . . . 29

3 Design of Micro-fluidic Cooling Configurations for 3D-ICs 33
3.1 Motivation of Micro-Fluidic Cooling . . . 33
3.2 Micro-channel Design Considerations/Constraints . . . 34
3.2.1 Cooling Power Consumption . . . 35
3.2.2 Non-uniform Power Profile . . . 35
3.2.3 TSV Constraint . . . 36
3.2.4 Thermal stress . . . 37
3.3 Hotspot Optimized Non-Uniform Micro-channel . . . 39
3.3.1 Problem Formulation . . . 40
3.3.2 Heuristic for Micro-channel Placement . . . 41
3.3.3 Workload-balanced Initial Micro-channel Distribution . . . 43
3.3.4 Micro-channel Cost Assignment . . . 49

3.4 TSV Constrained Bended Micro-channel 53
3.4.1 Motivation of Using Bended Micro-channel 53
3.4.2 Problem Formulation 54
3.4.3 Overall Micro-channel Design Flow 57
3.4.4 Mincost Flow Based Micro-channel Design 57
3.4.4.1 Initialization of Minimum Cost Flow Network 58
3.4.4.2 Cost Assignment 59
3.4.5 Micro-channel Refinement 62
3.4.5.1 Temperature and Pumping Power Analysis 62
3.4.5.2 Iterative Micro-channel Optimization 63
3.5 Hybrid Cooling Network 68
3.5.1 Motivation of Hybrid Cooling Network 68
3.5.2 Algorithm for Hybrid Cooling Network Design 70
3.5.3 Micro-channel Priority Assignment/Update 71
3.5.4 Thermal TSV Allocation and Sizing 72
3.5.4.1 Basic Thermal TSV Placement Approach 72
3.5.4.2 Modified Thermal TSV Allocation and Sizing Approach 74
3.5.4.3 Finding Maximum Independent Set $\mathcal{E}$ 75
3.6 Considering Thermal Variations 80
3.7 Cooling Performance of Micro-channel Designs 82
3.8 Runtime Thermal Management Using Micro-channels 83
3.8.1 Algorithm for Micro-fluidic Based DTM 84
3.8.2 Performance of Micro-channel Based DTM 87
3.9 Summary 88

4 Co-design of Electrical and Fluidic Cooling Systems 91
4.1 Motivation for Co-Design 91
4.2 Co-optimization of TSV Assignment and Micro-Channel Placement . 93
4.2.1 Problem Formulation 95
4.2.2 Algorithm for TSV Assignment and Micro-channel Placement Co-optimization 95
4.2.2.1 Overall Design Flow 95
4.2.2.2 Multi-commodity Minimum Cost Flow Formulation . 97
4.2.2.3 Iterative Optimization 102
4.2.3 Computational Simplifications 103
4.2.3.1 Multi Layer Case 103
4.2.3.2 Two Layer Case 105
4.2.4 Performance of TSV Assignment and Micro-channel Placement Co-design 107
4.2.4.1 Comparison of Wirelength and Pumping Power . . . 107
4.2.4.2 Tradeoff Between Wirelength and Pumping Power . . 110
4.3 Co-optimization of Gate Sizing and Micro-Fluidic Cooling 111
4.3.1 Motivation of Simultaneous Gate Sizing and Micro-channel Distribution 111

4.3.2 Modeling of Gate Delay 113
4.3.3 Problem Formulation 115
4.3.4 Algorithm for Gate Sizing and Micro-channel Placement Co-optimization 117
4.3.4.1 Step 1: Ideal Heat Sink and Gate Size Co-optimization 120
4.3.4.2 Step 2: Micro-channel Distribution for Ideal Case . . 121
4.3.4.3 Step 3: Gate Size and Grid Temperature Refinement 123
4.3.4.4 Step 4: Micro-channel Distribution Refinement . . . 129
4.3.4.5 Step 5: Re-iteration and Stopping Criteria 130
4.3.5 Performance of Gate Sizing and Micro-channel Placement Co-design 130
4.3.5.1 Comparison of Power Consumption 131
4.3.5.2 Comparison of Circuit Delay 132
4.3.6 Power-Performance Tradeoff 132
4.4 Summary 134

5 Conclusion and Discussion 136
5.1 Conclusion 136
5.2 Future Work 137

Bibliography 140

List of Figures

1.1 Interdependency between Electrical, Thermal, Reliability and Cooling 8

2.1 Stacked 3D-IC with micro-channel cooling system 12
2.2 Control volume of fluid 13
2.3 (a)-(f) Two phase flow patterns, (g) Evaporation process in a channel 16
2.4 Comparison of single and two phase flow 17
2.5 (a) Laminar flow pattern, (b) Turbulent flow pattern, (c) Transitional flow pattern 18
2.6 Fluid in micro-channel with bends 18
2.7 RC network for 3D-IC thermal modeling 20
2.8 Micro-channel thermal model 21
2.9 Thermal resistive network of one 3D-IC layer with micro-channels 23
2.10 A 3D-IC grid with thermal TSV 24
2.11 Exponential leakage model versus quadratic leakage model 27

3.1 Micro-channel and TSV configuration 34
3.2 Pumping power versus chip power consumption 35
3.3 Thermal stress inside and surrounding TSV (a) when chip temperature is 100℃, (b) when chip temperature is 50℃(assuming stress free temperature is 250℃) . . 38
3.4 Potential locations of micro-channels: (a) uniform spreading of micro-channels, (b) workload-balanced micro-channel spreading 42
3.5 Example of formulating mincost flow network, (a) 3D-IC structure, (b) abstract grid graph, (c) minimum cost flow network 47
3.6 (a) Cost initialization, (b) Cost update 52
3.7 Example of silicon layer thermal profile with TSV and (a) straight, (b) bended micro-channels 54
3.8 Example of micro-channel infrastructure design using minimum cost flow 55
3.9 Micro-channel infrastructure design flow 57
3.10 Cost assignment 61
3.11 Examples of (a) unbalanced cooling demand, (b) different number of bends . . . 64
3.12 Example of pairwise cooling workload balance 66
3.13 Examples of bend elimination 68
3.14 Overall design flow of micro-channel and thermal TSV co-optimization 71
3.15 Change in interdependence region of a grid (a) after allocating or enlarging a thermal TSV, (b) after shrinking a thermal TSV 79
3.16 Flow chart of micro-channel placement 81
3.17 Comparison of Pumping Power 83
3.18 Runtime pressure drop control versus fixed pressure drop for (a) group L, (b) group M, (c) group H 89

4.1 Conventional chip design flow 92
4.2 Thermal profile of one 3D-IC layer, and an example of TSV and micro-channel allocation where TSVs constraint us from allocating micro-channels at hotspots . 94
4.3 Overall design flow of MCMCF based algorithm 98

4.4 3D-IC with potential TSV and micro-channel locations 100
4.5 Multi-commodity min-cost flow formulation 101
4.6 Computationally simplifying transformation for multi-layer case 105
4.7 Computationally simplifying transformation for two-layer case 106
4.8 Tradeoff between wirelength and pumping power 111
4.9 Overall design flow 119
4.10 Delay versus power tradeoff for benchmark 1 133

List of Tables

3.1 Comparison of pumping power 84

4.1 Problem formulation 96
4.2 Benchmark Information 108
4.3 Comparison between our approach, TSV first and channel first approach ($P_{pump}: W$, $WL: m$, temperature: ^{o}C) 108
4.4 Comparison of total power consumption (power: W, t_{cons}: ns) 132
4.5 Comparison of circuit performance (power: W, t_{cons}: ns) 133

Chapter 1

Introduction

Moore's law has predicted a spectacular exponential growth in chip performance. However, in recent years, such performance improvements are slowing down, leading the research community to investigate alternative technologies that can restore the expected Moore's law rhythm in the functionality and cost of electronic products.

The three dimensional integrated circuit (3D-IC), which contains two or more layers of active electronic components that are stacked vertically, has become a significant technology for achieving continued performance improvements. The 3D-IC allows a significant increase in device densities, as well as faster on-chip communications compared with equivalent 2D circuits due to the shortening of interconnection length and increased bandwidth [30][88]. Besides the performance improvement, 3D-IC can also result in overall system energy savings and co-integration of heterogeneous components [22][49].

Despite these advantages, the 3D-IC also brings forth new challenges to chip thermal management due to the stacked structure.

1.1 Thermal Issues in 3D-ICs

Modern CPUs dissipate an average of $70 - 100W$ power while spatial and temporal power variations result in hotspots with even higher power density (up to $300W/cm^2$). The coming years will continue to witness a significant increase in CPU power dissipation due to advanced multi-core architectures and 3D integration technologies. Increase in CPU power density is usually accompanied by drastic increase in chip temperature. Nowadays the problems of increased chip power density, leakage power and system temperature have become major obstacles for further improvements in chip performance. The advent of 3D integration technology, exacerbates the thermal problems on chip since the power density increases dramatically due to several stacks of microprocessor chips, and also due to constraints imposed on heat flow paths (by several intermediate layers).

Recent data shows that more than 50% of all IC failures are related to thermal issues [58]. For instance, excessive temperature reduces the electron and hole mobilities which leads to increase in circuit propagation delay [44][83]; thermal variations and hotspots on chip cause reliability problems such as circuit mismatch and reduced chip lifetime (due to the cumulative damage caused by excessive temperature) [29][10][50]. Hence, loss of performance and reliability due to unpredictable thermal hotspots has become a major issue and limiting factor for further performance improvement in modern computer systems.

Furthermore, with continued scaling, the impact of leakage power is growing as well. Today, up to 50% (or even more) of the total power consumption is leakage

power [38]. It has been shown that leakage and temperature are highly interdependent: higher temperature increases the leakage power which in-turn further increases the temperature [47][80][27]. This interdependency increases the importance and difficulty of chip thermal management. The interdependence between temperature and leakage has been known for years and several attempts have been made during design time to better estimate/control the leakage and temperature through various design decisions [66][81]. For example [66] estimates the chip thermal and leakage profile while accounting for their interdependence, and [81] estimates the chip leakage profile while accounting for thermal variability.

In convectional computer systems, the thermal issues within the chip are handled at the package level by attaching a large heatsink on the top of the chip which dissipates heat into surrounding air, together with air cooling based cooling devices such as fans and air conditioners. Such "remote cooling" approaches have limitations in the following ways:

1. Fail to account for temporal variations: the processor operation exhibits great variations during runtime due to the nature of different applications and data. The demand for resources by different applications also varies. The processor operation and demand for resources influence the power and thermal states on-chip, hence the chip power and thermal profiles change during runtime. Therefore the convectional air cooling that ignores the real time chip operation and cooling demand is inefficient.

2. Fail to account for spatial variations: the chip power and thermal profiles also

exhibit variations spatially since different parts of the chip exhibit different switching activities. Such variations result in thermal hotspots which are important issues in electronic systems. The convectional heat sinks usually provide uniform cooling, which is very inefficient when there are hotspots.

3. Insufficient cooling capability: convectional heat sinks are usually attached at the top of the chip, which makes it ineffective in removing the heat inside the chip. Especially for 3D-ICs, the air based cooling has already been proved to be insufficient. As illustrated in [8], if two $100W/cm^2$ microprocessors are stacked on top of each other, the power density becomes $200W/cm^2$, which is beyond the heat removal limits of air cooled heat sinks.

Many efforts have been made to further mitigate the thermal issues in CPU chips. These efforts can be classified into three categories: CPU thermal management schemes [11][16][20][21][53][64][63], materials with better thermal property [67][79] and advanced cooling schemes [43][84][46][9]. In this work, we focus on the new cooling technology and dynamic thermal management for 3D-ICs.

1.2 Conventional Dynamic Thermal Management

Usually, the chip performance and temperature are closely related. In order to improve the performance delivered by the microprocessor, we could increase the transistor integration density of the chip, or increase the supply voltage and clock frequency, which leads to increased chip power consumption and temperature. Dynamic thermal management (DTM), where the chip operation is controlled during

runtime for curtailing thermal emergencies, can better address the temporal and spatial variations of the power and thermal profiles on-chip (in addition to the convectional package level cooling scheme). In conventional DTM schemes, thermal management can be achieved by controlling processor knobs such as core frequency and supply voltage [64][25][13][41], scheduling of tasks etc [93], which in effect, control the power dissipation in different parts of the chip. These schemes basically manage the chip temperature through controlling the heat dissipation rate/pattern. For example, in dynamic voltage and frequency scaling (DVFS), the supply voltage and operating frequency of micro-processors are dynamically controlled to reduce the chip power consumption, thereby reducing the temperature as well. However decreasing the supply voltage or operating frequency causes a potential performance reduction. Hence in the conventional DTM schemes, constraining the chip temperature is usually accompanied by reductions in performance.

With the continued application of conventional thermal management techniques, many of today's electronic systems underperform their inherent physical limits while operate at the highest power dissipation allowed by the available thermal management technology. CMOS, telecommunications, active sensing and imaging have undergone tremendous technological innovation over the last 40 years. However, despite the need and the potential for enhanced thermal management, electronic cooling technologies have changed very little in the past two or three decades, continuing instead to implement a "remote cooling" paradigm with only incremental improvements in performance.

1.3 Interlayer Micro-fluidic Cooling

Relying on the conventional air-cooled heat sink for the thermal management of 3D-ICs could have catastrophic consequences. On one hand, due to the strong thermal-performance interdependency, in order to limit on-chip temperatures, designers will resort to aggressive shutdown or slowdown resulting in significant underutilization of the available devices, hurting overall performance and leading 3D-ICs to experience greater fractions of dark silicon than that experienced by 2D-ICs. On the other hand, the heat removal challenge could limit the number of 3D layers or physical design optimization space. Consequently, if the performance and energy efficiency promised by 3D integration are to be realized, the thermal challenge needs to be actively addressed.

Micro-fluidic cooling, which integrates micro-channel heat sinks into silicon substrates of the chip and uses liquid flow to remove the heat from inside of the chip, can overcome this limitation. It has been reported to support heat dissipation higher than $700W/cm^2$[84]. Despite the excellent cooling capability, an overhead associated with micro-channel based heat removal technology is that the cooling system needs to consume extra energy for pumping the coolant through the channels. This has motivated a body of work that attempts to improve the micro-channel cooling effectiveness (thereby reducing the cooling energy consumption) by: a) controlling their dimensional parameters such as channel width, height and aspect ratio [42][84], b) investigating more sophisticated micro-channel infrastructures such as cross-linked micro-channels [32], micro-pin-fins [52][59], tree- or serpentine-shaped

micro-channels [68][23], and c) using hotspot optimized micro-channel structures [12][76], etc. Recently, micro-channel cooling has also been adopted in dynamic thermal management to control the runtime CPU performance and chip temperature by tuning the fluid flow rate through channels [19][18].

1.4 Interdependency between Electrical, Thermal, Reliability and Cooling

The electrical, thermal, reliability and cooling aspects of 3D-ICs are all interdependent. As the plot in Figure 1.1 shows, higher performance usually leads to greater chip power consumption and generates heat. Increase in chip temperature has a lot of detrimental effects.

1. It will result in higher circuit delay or delay uncertainties, which in turn limits the performance improvement.

2. Due to the interdependency between temperature and leakage power, increase in chip temperature will further increase the power consumption.

3. High chip temperature also exacerbate the electro-migration which will cause reliability loss.

On the other hand, the heat level inside the chip also decides the micro-fluidic cooling system configuration, which in turn changes the temperature/power distribution (due to thermal power interdependence), thereby changing the circuit delay and chip lifetime. Furthermore, the existence of micro-fluidic cooling also causes

greater thermal gradients. Such thermal gradients and reduced chip temperature will cause greater thermal stress, which on one hand, might result in mechanical reliability issues such as crack formation, and on the other hand will change the transistor delay.

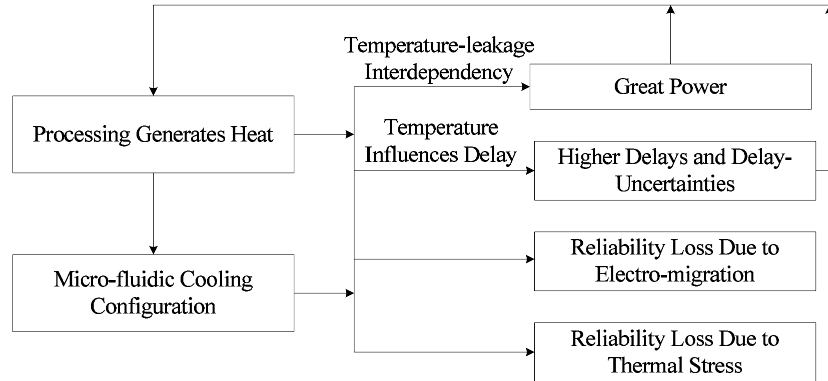


Figure 1.1: Interdependency between Electrical, Thermal, Reliability and Cooling

1.5 Advantage of Electrical and Cooling System Co-Design

In the conventional IC design flow, the electrical parts of the chip is designed first. The cooling system is then designed based on the current electrical system in place. However, due to the aforementioned interdependency, such design methodology (that separates electrical and cooling system design) is inefficient. Co-design of electrical and fluidic cooling system is necessary. It has the following advantages:

1. Higher cooling in timing critical areas results in better performing designs since transistor delay is proportional to temperature.

2. Higher cooling in timing critical areas enables us to aggressively pursue high power dissipating performance enhancements such as increasing supply volt-

age. This results in higher performance without impacting temperature since the extra heat can be managed by micro-fluidics.

3. The design optimization could be more aggressive since temperature issue can be addressed by aggressive cooling (placement, floorplanning etc).

4. Increasing the cooling levels in high leakage areas helps reduce the overall power since leakage is a highly non-linear function of temperature. Reduction in leakage may be significant enough to make increase in pumping power irrelevant.

5. Micro-fluidics may impact silicon thickness causing TSV performance degradation. By smart electrical design, this degradation could potentially be removed. For example, degradation in TSV performance could be overcome by stronger drivers.

1.6 Thesis Outline

In this work, we investigate optimization of micro-fluidic cooling system that can provide sufficient cooling to the 3D-IC with minimum overhead, while at the same time, addressing the design constraints imposed by the 3D-IC structure. Three micro-fluidic cooling configurations are proposed: hotspot-optimized non-uniform micro-channel, bended micro-channel and hybrid cooling network.

In order to fully explore the interdependency among electrical, thermal, reliability and cooling aspects of 3D-ICs, we also investigate electrical and micro-fluidic

co-design methodologies. With the co-design, fundamental power-performance improvements can be achieved.

This dissertation is organized in five chapters. Following this introduction is the background about 3D-IC and micro-fluidic cooling. In that chapter, we briefly introduce the fundamentals of micro-fluidic cooling, as well as thermal and power modeling of 3D-IC with micro-fluidic cooling. Chapter 3 discusses the design considerations of micro-fluidic cooling in 3D-ICs and presents three micro-channel heat sink configurations that addresses these considerations. A micro-fluidic cooling based dynamic thermal management (DTM) scheme is proposed. In Chapter 4, we investigate the electrical and cooling system co-design methodology. In that chapter, we focus on two aspects of the co-design: a) TSV assignment and micro-channel placement co-optimization, and b) gate sizing and micro-channel co-optimization. Finally, we conclude in Chapter 5 with a summary of the main findings of this work, and consider further prospects of this research field.

Chapter 2

Background

2.1 Basics of Three Dimensional Integrated Circuit

The 3D-IC contains two or more layers of active electronic components which are stacked vertically. Figure 2.1 shows a three-tiered stacked 3D-IC. In the 3D-IC, each active layer contains the functional units such as cores and caches, etc. The metal layer contains wires that enable communication among different components. There is also a metal-oxide layer above each metal layer. Through-silicon-vias (TSVs) are inserted in 3D-IC to deliver signal/power/ground among different tiers.

In 3D-IC, since several layers of electronic components that dissipates power are stacked vertically, its power density is usually higher than 2D-ICs, leading to potential thermal issues. Moreover, the thermal conductivity of oxide layer is low and hence would reduce the heat transfer towards the ambient. This exacerbates the thermal problems in 3D-ICs. Hence an important issue with 3D-IC is the removal of high density heat resulting from several stacks of microprocessor chips. Although current 3D-IC designs are limited to partitioning of memory and datapath across layers, future 3D-IC designs are expected to have significantly complex architectures and integration levels that would be associated with very high power dissipation and heat density.

In order to alleviate the thermal issues, micro-channel based liquid cooling and

thermal TSVs have been adopted. As shown in Figure 2.1, micro-channel heat sinks are embedded below the active layers. Liquid is pumped through each channel, and takes away the heat generated in the active layers [39][43]. The heated coolant is then cooled down in the heat exchanger, and recirculates into the fluid pump again for the cooling in the next circulation. On the other hand, TSVs, which are usually made of copper and have better thermal conductivity than silicon or metal-oxide, can help improve conduction of heat between different layers. When the number of signal TSVs is not enough, dummy thermal TSVs are inserted to further mitigate the thermal issues.

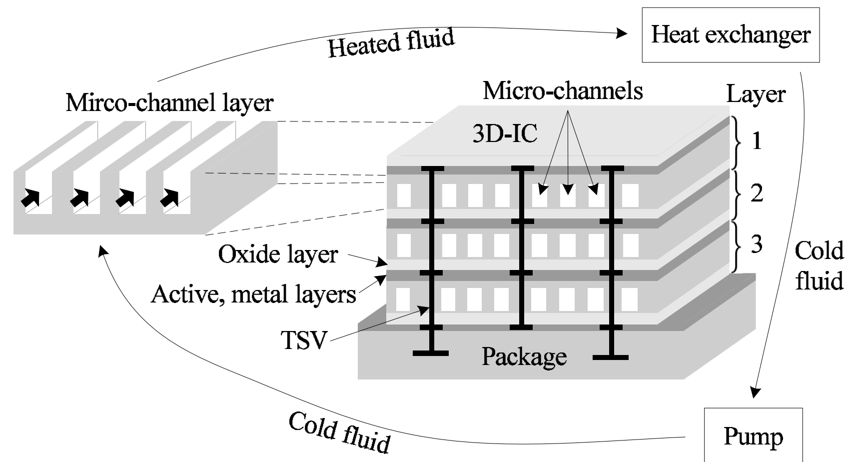


Figure 2.1: Stacked 3D-IC with micro-channel cooling system

2.2 Fundamental Characteristics of Fluids in Micro-channels

2.2.1 Conservation Law of Fluid Dynamics

The characteristic of fluid inside the micro-channels is governed by conservation law of fluid. Considering the control volume of fluid U and its surface S

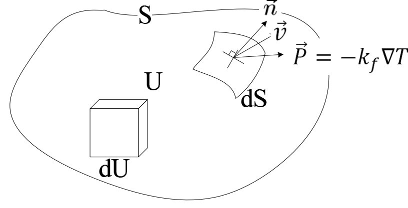


Figure 2.2: Control volume of fluid

(as shown in Figure 2.2). The fluid flow in the control volume is governed by the following mass, momentum and energy conservation equations [87][62][85][37][78]:

$$\begin{aligned} &\text{Mass conservation}: \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0 \\ &\text{Momentum conservation}: \rho(\frac{\partial \vec{v}}{\partial t} + \vec{v} \cdot \nabla \vec{v}) = -\nabla p + \mu \nabla^2 \vec{v} \\ &\text{Energy conservation}: C_v \frac{dT}{dt} + \nabla \cdot (-k_f \nabla T) + C_v \vec{v} \cdot \nabla T = \dot{P} \end{aligned} \tag{2.1}$$

Here $\vec{v}$ is the flow velocity vector, T is the fluid temperature, $\dot{P}$ is the volumetric heat generation rate, and p is the pressure inside fluid. Also, ρ, μ, C_v and k_f are the density, viscosity, volumetric specific heat and thermal conductivity of the fluid, respectively.

2.2.2 Dimensionless Numbers in Fluid Mechanics

The governing equations above are complex partial differential equations (PDE). Researchers in fluid mechanics introduced a set of dimensionless numbers which could help simplify the complex problem and also better understand the relative importance of forces, energies, or time scales [87][55]. Some of these dimensionless numbers are Reynolds number (Re), Prandtl number (Pr) and Nusselt number (Nu), etc.

Reynolds number Re: The Reynolds number gives a measure of the ratio between inertial forces to viscous forces, and is defined as:

$$Re = \frac{\rho v L_c}{\mu} \tag{2.2}$$

where v is the mean fluid velocity and L_c is the characteristic length. In straight micro-channels, the characteristic length is usually given by the hydraulic diameter D_h. When the cross section of the channel is circular, D_h is the diameter of the cross section, while in rectangular channels, D_h is defined as $D_h = 4 \cdot cross\ sectional\ area/perimeter = 4\Delta x \Delta z/(2\Delta x + 2\Delta z)$, where Δx and Δz are the width and height of the micro-channel. Usually, the Reynolds number is used to distinguish between laminar and turbulent flow, which will be explained later.

Prandtl number Pr: The Prandtl number is the ratio of momentum diffusivity (kinematic viscosity) to thermal diffusivity.

$$Pr = \frac{\text{kinematic viscosity}}{\text{thermal diffusivity}} = \frac{\mu/\rho}{k_f/(\rho C_v)} = \frac{C_v \mu}{k_f} \tag{2.3}$$

Nusselt number Nu: The Nusselt number is the ratio of convective to conductive heat transfer across the boundary between the fluid and solid. The Nusselt number is defined as:

$$Nu = \frac{h L_c}{k_f} \tag{2.4}$$

where h, L_c and k_f are the convective heat transfer coefficient, channel characteristic

length and fluid thermal conductivity. Usually, N_u is used to calculate the convective heat transfer coefficient h. Many works have been done to characterize the Nusselt number in micro-channels, and express it as a function of the Reynolds number and Prandtl number [15][5][89][94].

2.2.3 Single and Two Phase Flow

The working fluid in the micro-channel can be either single phase or two phase. The single phase flow consists of exclusively liquid coolant as the working fluid, while two phase flow consists of both liquid and vapor.

When the power density is too high so that the liquid absorbs too much heat and its temperature increases dramatically, part of the liquid will become vapor and two phase flow is formed. The two phase flow exhibits different patterns. Figure 2.3(a)-(f) shows the two phase flow patterns in horizontal channels. When the flow rate is low, the flow usually exhibits bubbly (Figure 2.3(a)) or plug pattern (Figure 2.3(b)), as the flow rate increases, the pattern becomes stratified (Figure 2.3(c)) and wavy (Figure 2.3(d)), and finally slug (Figure 2.3(e)) and annular (Figure 2.3(f)) [82][51].

The evaporation process in a channel is as Figure 2.3(g) shows. As the single phase liquid absorbs heat so that the temperature increases to the evaporation point, small bubbles appear. When the fluid continues to absorb heat along the channel, plug and slug flows appear. The flow becomes waved and annular in the end.

Figure 2.4 compares the cooling effectiveness of single and two phase flows.

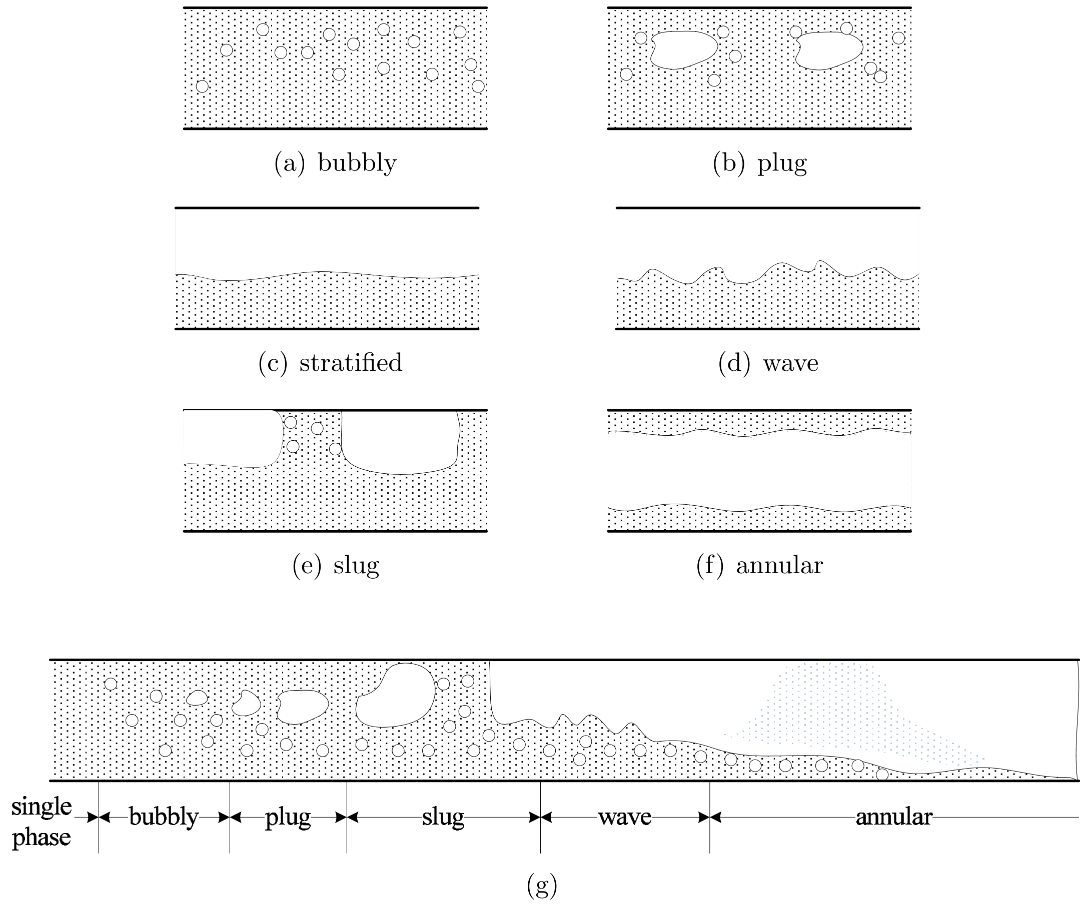


Figure 2.3: (a)-(f) Two phase flow patterns, (g) Evaporation process in a channel

It plots the solid temperature at the micro-channel outlet location T_w versus the footprint power density P_a for both single and two phase flows at same pumping power [6]. It shows that two phase flow achieves lower solid temperature than single phase flow, which indicates that two phase flow has higher cooling effectiveness than single phase flow.

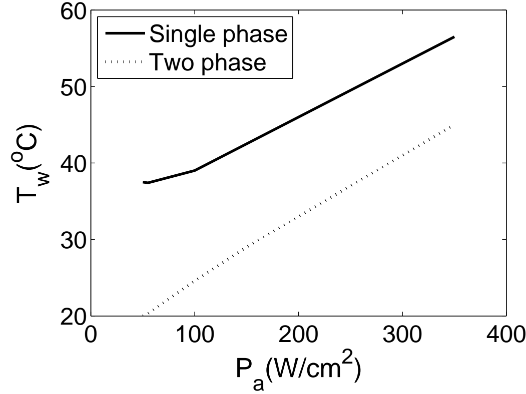


Figure 2.4: Comparison of single and two phase flow

2.2.4 Laminar and Turbulent Flow

The flow inside micro-channels can be laminar, turbulent, or transitional [55]. Figures 2.5(a), 2.5(b), 2.5(c) show these three types of patterns. Laminar flow (Figure 2.5(a)) occurs when fluid flows in parallel layers, with no disruption between the layers. That is, the pathlines of different particles are parallel. It generally happens in small channels and low flow velocities. In turbulent flow (as shown in Figure 2.5(b)), vortices and eddies appear, and make the flow unpredictable. Turbulent flow generally happens at high flow rates and larger channels. Transitional flow (Figure 2.5(c)) is a mixture of laminar and turbulent flow, with turbulence in the center of the channel, and laminar flow near the edges.

Usually, Reynolds number is used to predict the type of flow (whether laminar, turbulent or transitional) in straight channels. For example, as [55] shows:

When $Re < 2100$, it is laminar flow; when $2100 < Re < 4000$, it is transitional flow; when $Re > 4000$, it is turbulent flow.

When the channel involves more complex structure, the fluid exhibits more

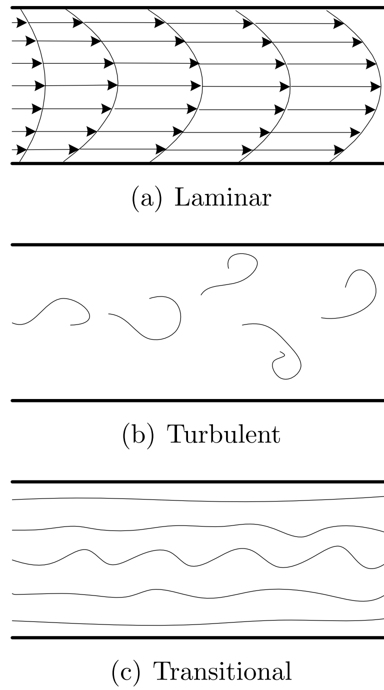


Figure 2.5: (a) Laminar flow pattern, (b) Turbulent flow pattern, (c) Transitional flow pattern

complicated behavior. Figure 2.6 shows an example of otherwise laminar flow in straight channels in a micro-channel with bends. When fluid enters a channel, it firstly subjects to a flow development process and after traveling some distance downstream, it becomes fully developed laminar flow. Then, when the flow comes across a bend, it becomes turbulent/developing around the corner and settles down after traveling some distance downstream into laminar fully developed flow again [68].

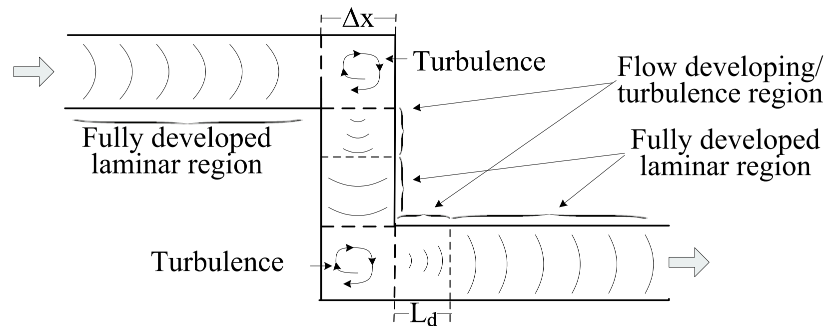


Figure 2.6: Fluid in micro-channel with bends

2.3 Thermal Modeling of 3D-IC with Micro-fluidic Cooling

2.3.1 Distributed RC Thermal Model

The chip thermal behavior can be modeled by a distributed RC network by partitioning it into fine grids. In this network, each grid is represented by a node. The voltage at each node represents the temperature at that grid. The current source in each grid represents the power dissipated at that location, so the chip power profile decides the current injected at each grid. Each resistance represents a heat transfer path between grids, while capacitors indicate the ability to store heat [77].

Figure 2.7 shows an example of the RC network for one 3D-IC layer. In this network, $R_{i,j}(i, j = 1...6)$ indicates the heat path (thermal resistance) between grids i and j, C_i represents the thermal capacitance of grid i. According to the thermal model, the thermal dynamics of each grid i is governed by the following equation:

$$\frac{dT_i}{dt} = -\sum_{\forall j \in \mathcal{N}(i)} \frac{T_i - T_j}{R_{i,j} C_i} + \frac{P_i}{C_i}, \quad \forall \text{grids } i \tag{2.5}$$

Here T_i is the temperature of grid i, and P_i is the power consumption at this grid, $\mathcal{N}(i)$ represents the set of grids adjacent to grid i.

In some works, people are more interested in the steady state thermal behavior. In this case, the thermal model can be simplified as a resistive network that represent steady state chip thermal behavior. Hence the governing equation in Equation 2.5 can be simplified as a set of linear equations of temperature and power as Equation

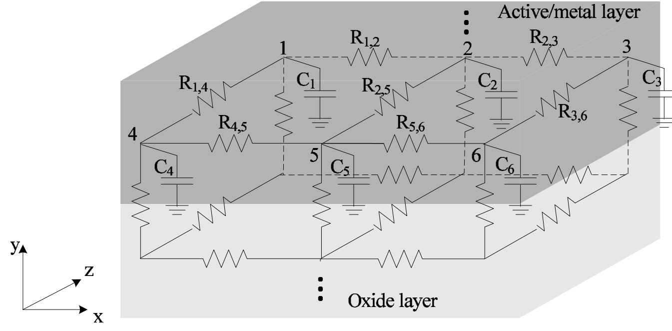


Figure 2.7: RC network for 3D-IC thermal modeling

2.6 shows. Given a chip thermal resistive network and power profile, the temperature profile can be estimated by solving the following system of linear equations.

$$\mathbf{G} \cdot \vec{T} = \vec{P} \tag{2.6}$$

Here $\mathbf{G}$ is the thermal conductance matrix decided by the thermal resistance network, $\vec{P} = \{P_i, \forall grid\, i\}$, $\vec{T} = \{T_i, \forall grid\, i\}$ represent the power and temperature profiles.

2.3.2 Cooling Performance of Micro-channels

The heat removal through micro-channels comprises of an intricate combination of heat conduction, convection and coolant flow. Consider the micro-channel in Figure 2.8, heat dissipated in surrounding regions (basically active layers) first conducts to the micro-channel sidewalls. The heat is then absorbed by the fluid through convection. The heated fluid is then carried away by the moving flow. These three aspects can be captured by expressing them as three types of thermal resistances: R_{cond} for conduction, R_{conv} for convection and R_{heat} captures fluid flow (as shown

in Figure 2.8).

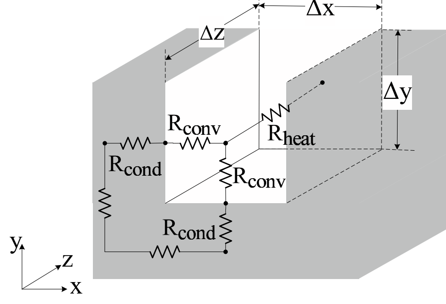


Figure 2.8: Micro-channel thermal model

Conductive resistance $\mathbf{R_{cond}}$: It is decided by thermal characteristics of the silicon that conducts heat dissipated in surrounding region to micro-channel sidewalls. It can be calculated using the model in [77].

Convective resistance $\mathbf{R_{conv}}$: It results from the convection of fluid, which moves the heat from micro-channel sidewalls to into the coolant fluid. The convective resistance depends on the fluid property and area for heat transfer between the micro-channel sidewalls and fluid. Assuming the micro-channel has been discretized into grids along the fluid direction z. The size of each grid is $\Delta x \times \Delta y \times \Delta z$ as Figure 2.8 shows. Let R_{conv} be the convective resistance between the micro-channel and sidewalls in each grid. As shown in [84], $R_{conv} = 1/hA_h$, where A_h is the surface area for heat transfer in each grid. If we assume that heat can be transferred from all four sidewalls, the surface area of each grid is $A_h = 2\Delta z(\Delta x + \Delta y)$. The parameter h is the coefficient of convective heat transfer explained in Section 2.2.2. Given the Nusselt number N_u and the micro-channel dimension, it is calculated by $h = N_u k_f / D_h$, k_f is fluid thermal conductivity and D_h is the hydraulic diameter. So the convective resistance could be expressed as:

$$R_{conv} = \frac{1}{hA_h} = \frac{D_h}{2N_u k_f \Delta z(\Delta x + \Delta y)} \tag{2.7}$$

Convective resistance $\mathbf{R_{heat}}$: The heat resistance basically represents the heat flowing downstream caused by the moving fluid:

$$R_{heat} = \frac{1}{C_v \rho f} \tag{2.8}$$

Here f is the volumetric flow rate in each channel. It depends on the fluid velocity v and micro-channel cross sectional area: $f = velocity * cross_sectional_area = v\Delta x\Delta y$. C_v is the fluid specific heat, and ρ is fluid density.

2.3.3 Overall Thermal Model of 3D-IC with Micro-channels

As indicated earlier, the thermal behavior of micro-channels can also be modeled by a thermal resistance network (Figure 2.8). The parameters of this resistive network could be computed using the equations described above or experiment based approaches [54]. The 3D-IC resistive network and micro-channel network can be combined to generate a unified model that captures the steady state thermal behavior of 3D-ICs with liquid cooling (Figure 2.9). Other aspects of 3D-ICs such as the thermal impact of TSVs and thermal wake effect [54] can also be incorporated in this resistive network.

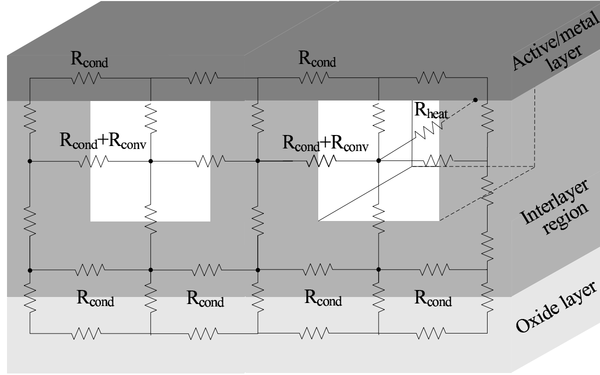


Figure 2.9: Thermal resistive network of one 3D-IC layer with micro-channels

2.3.4 Thermal Impact of TSVs

Besides of micro-fluidic cooling, some works have also proposed usage of dummy thermal TSVs for 3D-IC temperature reduction [24][17][90]. Due to the existence of oxide layer which separates different tiers of 3D-IC thermally, the heat cannot be effectively dissipated between tiers. The dummy thermal TSVs are firstly proposed by [14] as additional heat dissipation paths to alleviate the temperature issues on chip. Now it is adopted in 3D-ICs [24][17][90]. Since the TSV fill materials such as copper usually have much higher thermal conductivity than silicon and oxide, thermal TSVs could enhance the vertical heat transfer between different 3D-IC tiers and to the heat sinks by reducing the effective thermal resistances.

To quantify the thermal effect of TSV on 3D-IC, assuming there is a thermal TSV inserted in a 3D-IC grid as Figure 2.10 shows. The dimension of the grid is $\Delta x \times \Delta y \times \Delta z$, and its original vertical thermal conductivity is k_y^{old}. Assuming the cross sectional area of thermal TSV is A_{tsv}, the vertical thermal conductivity of this grid after inserting the thermal TSV becomes:

$$k_y^{new} = k_{tsv} \frac{A_{tsv}}{\Delta x \Delta z} + k_y^{old}(1 - \frac{A_{tsv}}{\Delta x \Delta z})$$
$$= k_y^{old} + (k_{tsv} - k_y^{old}) \frac{A_{tsv}}{\Delta x \Delta z} \tag{2.9}$$

Since the thermal conductivity of TSV fill material k_{tsv} is usually larger than the original thermal conductivity k_y^{old} (which is generally the thermal conductivity of silicon and metal-oxide), the thermal conductivity of this grid will increase after inserting the thermal TSV, which could result in better heat transfer between different tiers, and thus more uniform thermal profile.

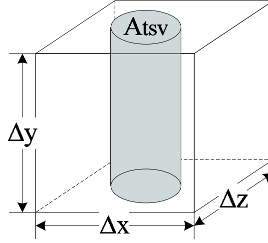


Figure 2.10: A 3D-IC grid with thermal TSV

2.4 Modeling of Power Consumption

The chip power consumption has two major components: dynamic power and leakage power [86]. Dynamic power results from charging of transistor load capacitances when they are switched, while leakage power is the power consumed by transistors when they are in idle state.

At the system level, there are generally three power states. A) Active mode, where the system is performing some operation. In this mode, the chip dissipates both dynamic power and leakage power. B) Standby mode, where the system is idle but ready to execute an operation. In this mode, the circuit dissipates only

leakage power. C) Inactive mode, where the power supply to circuits are shut down by power gating or other leakage reduction techniques. Very small amount of power is dissipated in this mode.

In addition to the chip power consumption, the micro-fluidic heat sink also consumes extra power for performing chip cooling. This power basically comes from the pump to inject the coolant through micro-channels. This extra cooling power consumption is called "pumping power".

2.4.1 Dynamic Power Consumption

The dynamic power depends on the transistor load capacitances being charged, the rate of switching, supply voltage, etc [86]. For each gate g_i, its dynamic power can be calculated by $P_{d,i} = \alpha_i C_{d,i} V_{dd}^2 F$, where $C_{d,i}$ is the load capacitance of gate g_i, α_i is its average switching activity in each cycle and F is the clock frequency. The output capacitance $C_{d,i}$ is proportional to gate width s_i, hence the dynamic power can also be represented as a function of gate size and clock frequency as Equation 2.10 shows. In the equation, $\beta_{d,i}$ depends on the switching activity α_i and supply voltage V_{dd}, etc.

$$P_{d,i} = \alpha_i C_{d,i}(s) V_{dd}^2 F = \beta_{d,i} s_i F \quad (2.10)$$

2.4.2 Leakage Power Consumption

Current leaks through transistors even when they are turned off, resulting in leakage power consumption. There are three main components of leakage power: reverse biased junction leakage, sub-threshold leakage and gate oxide tunneling leakage [86].

The junction leakage and sub-threshold leakage increases with temperature while the gate leakage is rather insensitive to temperature. [47] models the leakage-temperature dependency as:

$$P_{l,i} = \beta_{l,1} T_i^2 e^{-\frac{\beta_{l,2}}{T_i}} + \beta_{l,3} \tag{2.11}$$

As shown in [91], the variation of $e^{-\frac{1}{T_i}}$ is very small in the normal range of chip operating temperature. Hence, some works also approximate the leakage model as a quadratic function of temperature as Equation 2.12 shows [91]. The quadratic fitting parameters $\varepsilon_{1,2,3}$ are obtained from the underlying model in [47]. We tested the accuracy of this quadratic model. Figure 2.11 shows that the quadratic model is very close to the exponential model given in [47].

$$P_{l,i} = \varepsilon_1 T_i^2 + \varepsilon_2 T_i + \varepsilon_3 \tag{2.12}$$

The leakage power is also a linear function of gate width s_i [86]. Hence the overall

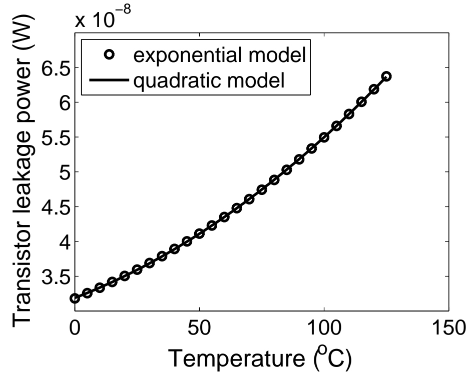


Figure 2.11: Exponential leakage model versus quadratic leakage model

leakage power can be modeled as (here φ is a constant):

$$P_{l,i} = \varphi \cdot s_i \cdot (\beta_{l,1} T_i^2 e^{-\frac{\beta_{l,2}}{T_i}} + \beta_{l,3}) \approx \varphi \cdot s_i \cdot (\varepsilon_1 T_i^2 + \varepsilon_2 T_i + \varepsilon_3) \tag{2.13}$$

From the power models, large gate size will result in higher dynamic and leakage power, which leads to temperature increase. Temperature increase in turn will lead to further increase in leakage power.

2.4.3 Micro-channel Cooling Power

2.4.3.1 Straight Micro-channels

The power used by micro-channels for performing chip cooling comes from the work done by the fluid pump to push the coolant fluid into micro-channels. It is a strong function of the level of heat removal desired. Basically, to maintain acceptable thermal levels, increase of chip power dissipation would result in increased pumping power P_{pump}, which is decided by the pressure drop and coolant fluid flow rate.

$$P_{pump} = \sum_{n=1}^{N} f_n \Delta p_n \tag{2.14}$$

Assuming there are N micro-channels, Δp_n and f_n are the pressure drop and fluid flow rate of the n-th micro-channel.

Here we assume the flow is fully developed laminar flow. The pressure drop in a micro-channel is decided by:

$$\Delta p = \frac{2\gamma\mu L v}{D_h^2} \tag{2.15}$$

where L is channel length, D_h is hydraulic diameter, v is fluid velocity, μ is the viscosity of fluid and γ is a function of micro-channel aspect ratio ($\frac{\Delta y}{\Delta x}$) [42][39].

In this work, we assume that all straight micro-channels have the same width and height. Usually fluid pumps are designed to work such that all the micro-channels experience the same pressure drop Δp. For a given pressure drop that the pump delivers across all channels, fluid velocity v could be estimated using Equation 2.15. So the fluid flow rate $f = v\Delta x\Delta y$ is also a function of pressure drop Δp, and could be estimated. Since the pressure drop is the same across all channels, so are the velocity and fluid flow rate since we assume all channels have the same dimension. Given this, the pumping power can be rewritten as:

$$P_{pump} = Nf\Delta p = \frac{N\Delta x\Delta y D_h^2 \Delta p^2}{2\gamma\mu L} \tag{2.16}$$

2.4.3.2 Micro-channels with Bends

Consider the micro-channel structure shown in Figure 2.6. The existence of a bend causes a change in the flow properties which impact the cooling effectiveness and pressure drop. An otherwise fully developed laminar flow in the straight part of the channel, when comes across a 90° bend becomes turbulent/developing around the corner and settles down after traveling some distance downstream into laminar fully developed again (see Figure 2.6). So a channel with bends has three distinct regions, 1) fully developed laminar flow region, 2) the bend corner, and 3) the developing/turbulent region after the bend [33][68]. The length of flow developing region is [69]:

$$L_d = (0.06 + 0.07\frac{\Delta y}{\Delta x} - 0.04\frac{\Delta y^2}{\Delta x^2})ReD_h \tag{2.17}$$

where Re is the Raynolds number, and Δx, Δy and D_h are the micro-channel width, height and hydraulic diameter.

The rectangular bend impacts the pressure drop. Due to the presence of bends, the pressure drop in the channel is greater than an equivalent straight channel with exactly the same dimensions. The total pressure drop in a channel with bends is the sum of the pressure drop in the three regions described above (which finally depend on how many bends the channel has). Assume L is the total channel length, and m is the bend count. Therefore $m \cdot L_d$ is the total length that has developing/turbulent flow and $m \cdot \Delta x$ is the total length attributed to corners (see Figure 2.6). Hence the effective channel length attributed to fully developed laminar flow is $L - m \cdot L_d - m \cdot \Delta x$. The pressure drop in the channel is the sum of the pressure

drop in each of these regions.

Pressure drop in fully developed laminar region: The total pressure drop in fully developed laminar region is [42]:

$$\Delta p_f = \frac{2\gamma\mu(L - m \cdot L_d - m \cdot \Delta x)v}{D_h^2} = \frac{2\gamma\mu L_f v}{D_h^2} \tag{2.18}$$

Here $L_f = L - m \cdot L_d - m \cdot \Delta x$ is the total length of the fully developed laminar region which is explained above, the other parameters are the same as in Equation 2.15.

Pressure drop in flow developing region: The pressure drop in each flow developing region is: $\delta p_d = \frac{2\mu v}{D_h^2} \int_0^{L_d} \psi(z)dz$ [56]. Here $\psi(z)$ is given by $\psi(z) = 3.44\sqrt{(ReD_h)/z}$, where z is the distance from the entrance of developing region in the flow direction. Assuming there are a total of m corners in a given micro-channel, so there are m developing regions with the same length L_d in this channel. By putting the expression of $\psi(z)$ and L_d into the equation of δp_d and solving the integration, we can get the total pressure drop of the developing region in this micro-channel:

$$\Delta p_d = m \cdot \delta p_d = mK_d\rho v^2 \tag{2.19}$$

where $K_d = 13.76(0.06 + 0.07\frac{\Delta y}{\Delta x} - 0.04\frac{\Delta y^2}{\Delta x^2})^{\frac{1}{2}}$ is a constant associated with the aspect ratio $\frac{\Delta y}{\Delta x}$. Please refer to [33][56] for details.

Pressure drop in corner region: The total pressure drop at all the 90° bends in a micro-channel is decided by:

$$\Delta p_{90^\circ} = m \cdot \delta p_{90^\circ} = m\frac{\rho}{2}K_{90}v^2 \tag{2.20}$$

where m is the number of corners in the channel, δp_{90° is the pressure drop at each bend corner and K_{90} is the pressure loss coefficient for 90° bend whose value can be found in [33].

Total pumping power: The total pressure drop in a micro-channel with bends is the sum of the pressure drop in the three regions discussed above:

$$\Delta p = \Delta p_d + \Delta p_f + \Delta p_{90^\circ} = \frac{2\gamma\mu L_f}{D_h^2}v + m(K_d + \frac{K_{90}}{2})\rho v^2 \tag{2.21}$$

From Equations 2.21, the total pressure drop of a micro-channel is a quadratic function of the fluid velocity v. For a given pressure difference applied on a micro-channel, we can calculate the associated fluid velocity by solving Equation 2.21. With the fluid velocity, we can then estimate the fluid flow rate f, and thus estimate the thermal resistance and pumping power for this channel. Hence the pumping power as well as cooling effectiveness of micro-channels with bends is a function of 1) number of bends, 2) location of channels, and 3) pressure drop across the channel.

Comparing Equations 2.15 and 2.21, due to the presence of bends, if the same

pressure drop is applied on a straight and a bended micro-channel of the same length, the bended channel will have lower fluid velocity, which leads to a lower cooling capability. Therefore, to provide the same amount of cooling, we will need to increase the overall pressure drop that the pump delivers, which results in increase of pumping power. But bends allow for better coverage in the presence of TSVs.

Chapter 3

Design of Micro-fluidic Cooling Configurations for 3D-ICs

3.1 Motivation of Micro-Fluidic Cooling

The coming years will witness a significant increase in CPU power dissipation due to advanced multi-core architectures and 3D integration technologies. The thermal problem in 3D-IC is even more severe compared with 2D circuits, because the power density is usually higher due to the stacked architecture. Moreover, the thermal conductivity of oxide layer is low and hence would reduce the heat conduction towards the ambient. The conventional air cooling has been proved to be insufficient for future high performance 3D-ICs even with sophisticated DTM schemes [8]. As a result, more effective active cooling schemes are being investigated for high performance 3D-ICs [39][43]. Micro-channel cooling, which integrates micro-channel heat sinks into each tier of the 3D-IC and uses liquid flow to remove heat from within the 3D chip, is an effective active cooling scheme for 3D-IC. It has been reported to support heat dissipation higher than $700W/cm^2$ with single phase flow[84]. When the working fluid is two phase flow, the heat removal rate is even higher.

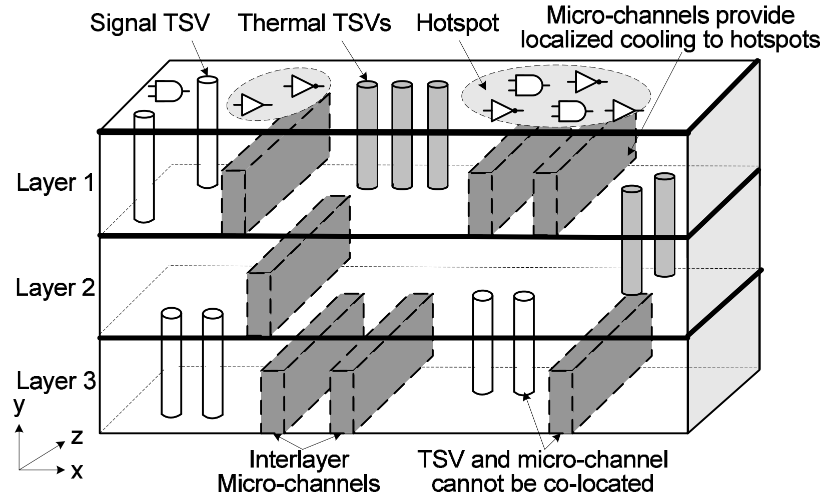


Figure 3.1: Micro-channel and TSV configuration

3.2 Micro-channel Design Considerations/Constraints

As shown in Figure 2.1, each tier of 3D-IC contains an active silicon layer and silicon substrate. The micro-channels are placed horizontally in the silicon substrate. TSVs such as power/ground TSV, signal TSV, etc, are incorporated for communications between layers and delivery of power and ground. Figure 3.1 shows a possible configuration of micro-channels and TSVs in the silicon substrate of 3D-IC [40][45]. In each 3D-IC tier, micro-channels are etched in the inter-layer region (silicon substrate). Fluidic channels (fluidic TSVs) go through all the tiers and delivers coolant to micro-channels. TSVs also go through the silicon substrate vertically to deliver signal, power and ground.

Though the micro-channel heat sink is capable of achieving good cooling performance, many problems need to be addressed when designing the micro-channel infrastructure for cooling 3D-IC so as to ensure the reliability of the chip and also improve the effectiveness of the micro-channel [72].

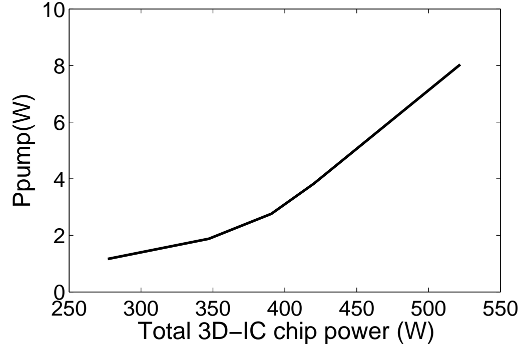


Figure 3.2: Pumping power versus chip power consumption

3.2.1 Cooling Power Consumption

The micro-fluidic cooling is active by nature. That is, the fluid pump consumes extra energy for pushing the coolant through the micro-channels (we call this *pumping power* consumption). The pumping power can be quite significant. Figure 3.2 shows the pumping power required to maintain the 3D chip below temperature constraints (85℃) for different chip power profiles using the conventional approach of spreading straight micro-channels all over each tier. For each power profile, we find the minimum pressure drop required to maintain the chip temperature within constraints and then estimate the pumping power under this pressure drop using Equation 2.16. As we can see, to maintain the chip temperature within acceptable levels, pumping power increases very fast as the total chip power increases. Therefore controlling the micro-channel pumping power is very important.

3.2.2 Non-uniform Power Profile

The underlying heat dissipated in each active silicon layer exhibits great non-uniformity [39][60]. Such non-uniformity in power profile results in hotspots in

thermal profiles. Therefore, when designing micro-channel heat sink infrastructure, one should account for this non-uniformity in thermal and power profiles. Simply minimizing the total equivalent thermal resistance of the micro-channels while failing to consider the non-uniformity of the power profile will lead to suboptimal design. For example, conventional approaches for micro-channel designs spread the entire surface to be cooled with channels, and find the width and height of micro-channels that minimize the overall thermal resistance [84][42]. This approach, though helps reducing the peak temperature around the hotspot region, over cools areas that are already sufficiently cool. This is wasteful from the point of view of pumping power.

3.2.3 TSV Constraint

3D-ICs impose significant constraints on how and where the micro-channels could be located due to the presence of TSVs, which allow different layers to communicate. As illustrated in Figure 3.1, micro-channels are allocated in the interlayer bulk silicon regions. TSVs also exist in this region, causing a resource conflict. A 3D-IC usually contains thousands of TSVs which are incorporated with clustered or distributed topologies [26][57]. These TSVs form obstacles to the micro-channels since the micro-channels cannot be placed at the locations of TSVs. Therefore the presence of TSVs limits the available spaces for micro-channels, and designing the micro-channel infrastructure should take this fact into consideration.

3.2.4 Thermal stress

The TSV fill materials are usually different from silicon. For example, copper has low resistivity and is therefore widely used as the material for TSV fill. Because the annealing temperature is usually much higher than the operating temperature, thermal stress will appear in silicon substrate and TSV after cooling down to room temperature due to the thermal expansion mismatch between copper and silicon [92][7]. This thermal stress might result in reliability problems such as cracking.

Moreover, as shown in [92][28], thermal stress also influences electron/hole mobilities significantly, hence changing the gate delay. Therefore, if the gates on critical paths are allocated near TSVs (basically regions with high thermal stress), timing violation might occur.

The existence of micro-channels which influences the temperature around TSVs will influence the thermal stress, thereby changing the mechanical reliability analysis and timing analysis in the 3D-IC with TSVs. For example, Figure 3.3 shows the thermal stress inside and surrounding a TSV at different thermal conditions. Figure 3.3(a) depicts the thermal stress when chip temperature is 100℃ and annealing temperature (which is basically the stress free reference temperature) is 250℃. The figure shows that large thermal stress (up to 490MPa) appears surrounding the TSV. Figure 3.3(b) depicts the thermal stress when the chip temperature is 50℃. In this case (where chip temperature is 50℃), the overall thermal stress is increased (compared with the previous case where chip temperature is 100℃), and the maximum thermal stress reaches up to 670MPa. Such phenomenon indicates

that reduction in chip temperature results in an increase in thermal stress. Hence the existence of micro-channels, which generally reduces chip temperature, may increase the TSV-induced thermal stress. Such phenomenon should be considered when designing the micro-channel infrastructure.

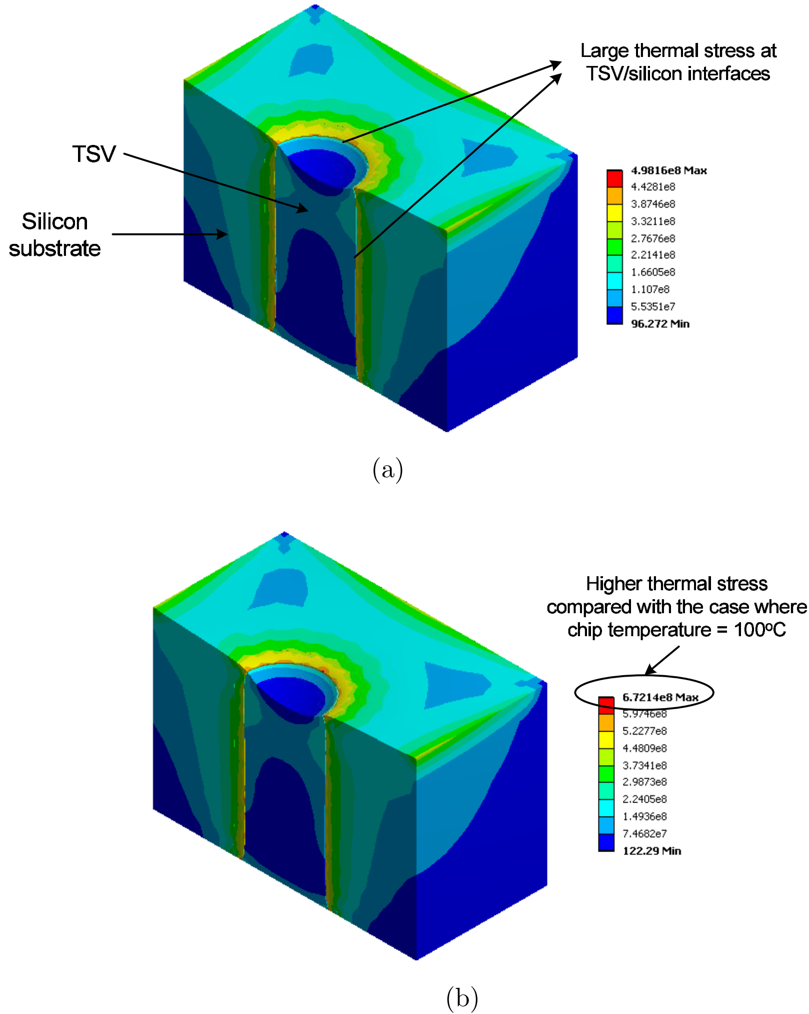


Figure 3.3: Thermal stress inside and surrounding TSV (a) when chip temperature is 100℃, (b) when chip temperature is 50℃(assuming stress free temperature is 250℃)

Moreover, if micro-channels are placed too close to the TSVs, the silicon walls between the TSVs and micro-channels will be more likely to crack because the walls are thin. These facts further limits the locations of micro-channels.

In this chapter, we propose three micro-channel structures (cooling configura-

tions) to improve the cooling effectiveness while still satisfying the design constraints imposed on micro-channels. These three structures are: non-uniform (hotspot optimized) micro-channels [76], bended (TSV-constrained) micro-channels [73] and hybrid cooling network [75]. We also investigate a micro-channel based dynamic thermal management scheme that controls the runtime chip temperature by tuning the pressure drop (fluid flow rate) through micro-channels [71].

3.3 Hotspot Optimized Non-Uniform Micro-channel

The first configuration is hotspot-optimized non-uniformly distributed micro-channels [76]. In this work, we start from the regular straight micro-channels. According to the micro-channel thermal model in Section 2.3.2, the cooling effectiveness of micro-channels depends on the dimension and distribution of micro-channels, as well as the fluid flow rate through micro-channels. The pumping power required by micro-channels also depends on these parameters as Equation 2.16 shows. Here we assume the micro-channel width and height are fixed. The optimal micro-channel width and height were investigated in [84][42], etc. In this case, designing the optimal micro-channel structure is basically deciding the count and distribution of micro-channels. For a given pressure drop, increase in the number of micro-channels helps increasing the coverage of cooling system thereby improving the heat removal rate. But this will also lead to linear increase in total pumping power.

3.3.1 Problem Formulation

Given a 3D-IC design, its power distribution is a function of the architecture and application. Assuming the power profile is given (this assumption will be generalized later), and we know a set of locations as potential target locations for micro-channels (see Figure 3.4) (all locations containing TSVs have been removed from this set for the sake of illustration). We want to find the number and locations of channels such that the temperature all over the chip is within acceptable limits while minimizing the number of channels (assuming pressure drop Δp is fixed). The problem is formulated as follows:

$$\begin{aligned} unknowns: &\quad \mathbf{B} \\ min &\quad N = sum(\mathbf{B}) \\ s.t. &\quad \vec{T}(\mathbf{B}) \leq T_{max} \end{aligned} \tag{3.1}$$

Here, $\mathbf{B} = \{B_1, B_2, ..., B_N\}$ is a vector representing the locations of all micro-channels. Assuming we know the set of potential micro-channel locations, each element $B_n(n = 1...N)$ in $\mathbf{B}$ corresponds to one of these locations and it's value is assigned as:

$$B_n = \begin{cases} 1, \text{micro-channel exists in this location} \\ 0, \text{otherwise} \end{cases} \tag{3.2}$$

N is the total number of micro-channels placed. When pressure drop Δp is given, pumping power only depends on the total number of micro-channels N, so the objective in Equation 3.1 basically minimizes the pumping power. For a given

allocation of micro-channels, the thermal resistive network can be used to estimate temperature profile by dividing the 3D-IC into grids (using the approach illustrated in Section 2.3.3). Finding the optimal locations of micro-channels is a complex discrete problem. Now we describe an iterative heuristic that finds a good solution.

3.3.2 Heuristic for Micro-channel Placement

Algorithm 1 gives the basic framework of our heuristic. The heuristic is based on iterative improvement. We start by finding a set of potential locations for micro-channels as Figure 3.4 shows. Note that all the locations containing TSVs and other structures are removed from this set for the sake of illustration. In reality the potential location would be limited by TSV locations, etc. The detailed approach for finding the potential micro-channel locations is given in Section 3.3.3. In the initial micro-channel design, micro-channels are placed at all the potential locations. We assign each micro-channel a cost which represents the impact of removing the micro-channel on thermal profile. Given the initial design and micro-channel cost, the algorithm iteratively removes micro-channels until further removal results in thermal violation. In each iteration, the micro-channel with the smallest cost is removed. After each micro-channel removal, the costs of the remaining micro-channels need to be updated. This is because the impact of removing a micro-channel on the thermal profile is a function of both the power profile and also which micro-channels have been removed so far. A micro-channel that had little impact on the thermal profile if many micro-channels were present in its neighborhood might have a much

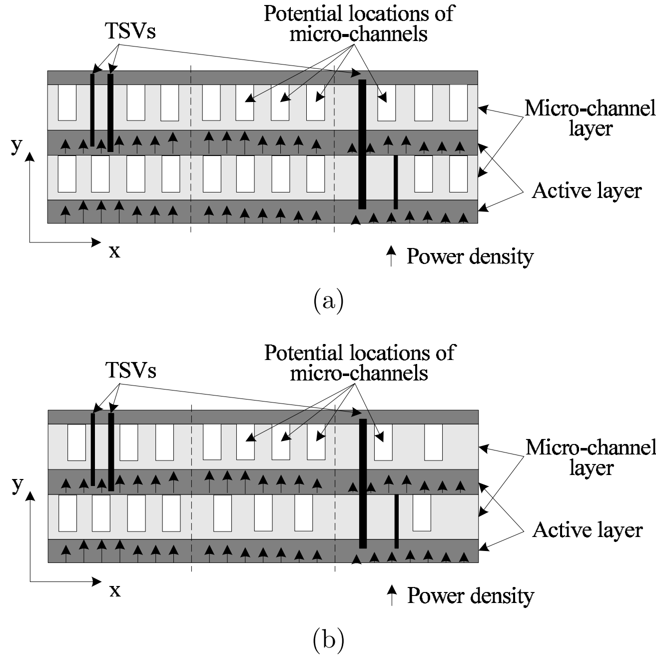


Figure 3.4: Potential locations of micro-channels: (a) uniform spreading of micro-channels, (b) workload-balanced micro-channel spreading

Algorithm 1 Heuristic for micro-channel placement

Starting from micro-channels placed at all potential locations:
1. Initialize the cost (defined below) for each micro-channel;
2. Set viscosity $\mu = \mu(T_{in})$, where T_{in} is coolant inlet temperature;
3. Repeat:
4. Remove micro-channel with the lowest cost;
5. Generate the new resistive thermal model;
6. Estimate the temperature profile $\vec{T}$
7. If $\vec{T} \leq T_{max}$, update cost and viscosity, and go to step 2;
8. Else stop.

higher impact when its neighboring micro-channels have been removed. Since the fluid viscosity μ is a function of fluid temperature, we also update the value of fluid viscosity after each iteration. To estimate the new viscosity, we calculate the average fluid temperature among all channels and lookup the associated viscosity from the table in [34].

The complexity in this optimization problem comes from the fact that as we change the location of channels, the underlying thermal resistive network changes.

In order to estimate the thermal impact, we need to solve Equation 2.6 every time we have a new resistive network, which, even though exhibits linear complexity for estimation of the thermal profile, can have high complexity due to the granularity of the grid. The success (both performance and runtime) of this algorithm critically depends on how potential micro-channel locations are distributed (which basically decides the initial micro-channel distribution) and how micro-channel cost is assigned and updated. In the next three subsections, we will discuss these aspects and investigate ways to improve the efficiency of the algorithm (basically reducing the required number of iterations in the algorithm).

3.3.3 Workload-balanced Initial Micro-channel Distribution

The heuristic of micro-channel placement starts with an initial distribution where micro-channels are placed at all the potential locations and iteratively removes micro-channels. The complexity of the algorithm mostly comes from the thermal estimation in each iteration. Hence we should reduce the number of iterations required by the algorithm (while still maintaining its performance), which critically depends on how potential micro-channel locations are distributed. So in this section, we investigate the method to find a good initial micro-channel distribution, which is basically finding a set of potential locations of micro-channels.

As shown in [39] and [60], the underlying heat dissipated in each active silicon layer exhibits great non-uniformity. For example, typical CPU designs are generally very hot in areas surrounding ALU and cooler around caches. Therefore, spreading

micro-channels all over the 3D chip or using arbitrary initial micro-channel distribution may result in imbalance in micro-channel cooling workloads and waste pumping power. For example, for a 3D-IC shown in Figure 3.4, in the active silicon layer which dissipates power, the height of the arrow indicates the power density. If the potential micro-channel locations spread the entire chip as Figure 3.4(a) shows, the regions with higher power density are covered by similar amount of micro-channels as lower power regions. Since all channels have the same pressure drop and dimension (therefore provides same cooling capability), in order to cool the higher power density region, we need to increase the pressure drop or dimension of all channels, which is unnecessary for low power regions and leads to waste of pumping power.

Therefore, we consider spreading the potential micro-channel locations according to the spatial variations in power/thermal profiles on chip. Intuitively, in those locations where the potential cooling workload is high, we try to place more micro-channels as Figure 3.4(b) shows. In other words, each micro-channel should absorb same/similar amount of heat. This initial distribution could then be further optimized by our iterative approach described earlier. The problem of finding the initial micro-channel distribution is formally stated as follows:

Problem Statement: Given a 3D-IC and a power profile, we would like to find N potential micro-channel locations in the micro-channel layers such that all the channels will absorb the same amount of heat. The amount of heat each micro-channel absorbs can be estimated as follows: assuming the 3D-IC is divided into grids and modeled as a thermal resistive network (as Figure 2.9). The heat absorbed by micro-channel (i, j) is:

$$P_{heat,i,j} = \sum_{\substack{\forall (i_2,j_2,k_2) \in \mathcal{G}(i,j) \\ \forall (i_1,j_1,k_1) \notin \mathcal{G}(i,j)}} I(i_1,j_1,k_1;i_2,j_2,k_2) \tag{3.3}$$

Here $I(i_1,j_1,k_1;i_2,j_2,k_2)$ is the heat (current) flowing from grid (i_1,j_1,k_1) to grid (i_2,j_2,k_2) (note that (i_1,j_1,k_1) and (i_2,j_2,k_2) must be neighboring grids), and $\mathcal{G}(i,j)$ is the set of grids covered by micro-channel (i,j) (micro-channel located at the i-th/j-th grid in x/y direction). Therefore (i_2,j_2,k_2) is a grid inside micro-channel (i,j) while (i_1,j_1,k_1) is outside micro-channel (i,j), and $I(i_1,j_1,k_1;i_2,j_2,k_2)$ indicates the heat flowing into micro-channel (i,j) through grids (i_1,j_1,k_1) and (i_2,j_2,k_2). Here $I(i_1,j_1,k_1;i_2,j_2,k_2)$ can be estimated by thermal analysis. For example, assuming the temperature at grids (i_1,j_1,k_1) and (i_2,j_2,k_2) are T_{i_1,j_1,k_1} and T_{i_2,j_2,k_2}, then $I(i_1,j_1,k_1;i_2,j_2,k_2) = (T_{i_1,j_1,k_1} - T_{i_2,j_2,k_2})/R^{i_2,j_2,k_2}_{i_1,j_1,k_1}$, where $R^{i_2,j_2,k_2}_{i_1,j_1,k_1}$ is the thermal resistance between grids (i_1,j_1,k_1) and (i_2,j_2,k_2) (it is usually a combination of convective and conductive resistances). Therefore $I(i_1,j_1,k_1;i_2,j_2,k_2)$ depends on the micro-channel structure (location and size). Assuming the total number of potential micro-channel locations N is fixed, we would like to allocate these N micro-channels so that the heat each micro-channel absorbs ($P_{heat,i,j}$) are the same.

The difficulty in this problem comes from the fact that the amount of heat each micro-channel absorbs is hard to decide before micro-channel placement, since the location of micro-channels and pressure drop will largely influence the direction of heat flow and thereby influence the heat each micro-channel absorbs. Therefore, we use a minimum cost flow based heuristic to find a good initial micro-channel

density distribution.

Formulation of minimum cost flow problem:

To form the minimum cost flow problem, we firstly divide the 3D-IC into *coarse grids* and each grid can contain several micro-channels. Basically we would like to decide the density distribution of the potential micro-channel locations among the grids. Finding the density distribution of micro-channels is basically deciding the number of micro-channels in each grid. Note that, since the micro-channel encompasses the whole chip in z direction, the number/location of micro-channels in the grids at same (x, y) position are the same. So we use $N_{i,j}$ to denote the number of channels in the i-th/j-th grids in x/y direction (note that the grid network is coarse). The density of micro-channels should be proportional to the potential cooling workload for the micro-channels in this region.

After dividing the 3D-IC into grids, we perform a thermal analysis based on this grid division assuming there is no micro-channel, and estimate the temperature at each grid. Meanwhile, we abstract the 3D-IC structure as an undirected graph. Figure 3.5 gives an example of how we form the minimum cost flow problem based on the given 3D-IC structure and thermal profile. Figure 3.5(a) shows a 3D-IC with two active silicon layers and a micro-channel layer in between. This 3D-IC is divided into coarse grids and an associated graph which captures the 3D-IC structure is formed in Figure 3.5(b) and the corresponding minimum cost flow problem is given in Figure 3.5(c). As we can see from Figure 3.5(b), each grid is represented by a node, and each pair of neighboring grids (nodes) are connected by an undirected

edge. Based on this graph and the temperature profile, the minimum cost flow problem is formed as follows:

Figure 3.5: Example of formulating mincost flow network, (a) 3D-IC structure, (b) abstract grid graph, (c) minimum cost flow network

Nodes: a) Each node (i, j, k) in the active silicon layer forms a source node, with $a_{i,j,k} = max\{0, T_{i,j,k} - T_{in}\}$ units of flow available, where $T_{i,j,k}$ is the temperature at grid (i, j, k) and T_{in} is the constant fluid inlet temperature. As shown in Figure 3.5(b), the active layer nodes are represented by black dots, and becomes source nodes in the minimum cost flow problem in Figure 3.5(c).

b) There is a single sink node with demand $\sum_{\forall (i,j,k) \in active\,layer} a_{i,j,k}$. This node is represented by a black square in the minimum cost flow in Figure 3.5(c).

c) Each of the other grids/nodes is represented by an intermediate node (gray dots in Figure 3.5(c)).

Edges: a) Similar as the graph in Figure 3.5(b), in the minimum cost flow in Figure 3.5(c), each pair of neighboring nodes are connected by an edge and the edges are bi-directional (can take heat flow in either direction). Each edge has unlimited capacity and also a cost which is assigned as:

$$cost(i_1, j_1, k_1; i_2, j_2, k_2) = r_1 \cdot (T_{i_1,j_1,k_1} + T_{i_2,j_2,k_2})/2 \tag{3.4}$$

Here $cost(i_1, j_1, k_1; i_2, j_2, k_2)$ denotes the cost of edge connecting nodes (i_1, j_1, k_1) and (i_2, j_2, k_2). The cost is basically decided by the average temperature of the two neighboring nodes, and r_1 is a constant scaling factor.

b) All the nodes in the micro-channel layers are connected to the sink node with the capacity and cost defined as follows:

$$\begin{aligned} capacity &: cap(i, j, k) = r_2 V - r_3 n_{TSV}^{i,j,k} \\ cost &: \quad cost(i, j, k; sink) = r_4 T_{i,j,k} \end{aligned} \tag{3.5}$$

Here V is a constant representing the maximum number of micro-channels each grid can contain, $n_{TSV}^{i,j,k}$ represents the number of TSVs in grid (i, j, k), r_2, r_3 are constant scaling factors and r_4 is a small constant. The edge capacity is decided by the number of micro-channels each grid could contain at most, which depends on the number of TSVs in the grid. The existence of TSVs in a grid would reduce the capacity of each grid since micro-channel cannot be placed in the places where there are TSVs.

The minimum cost flow problem basically sends the flows from source nodes to the sink node through some of the edges so that the total cost of the selected edges is minimized. The solution of minimum cost flow gives the amount of flow $(e_{i,j,k})$ that passes through each micro-channel layer node (i, j, k). Assuming N is the total number of potential micro-channel locations that we would like to find, the

number of micro-channels in grids $(i, j, \forall k)$ is assigned as follows:

$$N_{i,j} = round(\frac{\sum_{\forall k} e_{i,j,k}}{\sum_{\forall i,j,k} e_{i,j,k}} N) \tag{3.6}$$

The $round()$ function means rounding the fractional number to the nearest integer number. After getting the number of micro-channels in each grid, we uniformly place such amount of micro-channels in each grid. That is, $N_{i,j}$ micro-channels are uniformly distributed in grids $(i, j, \forall k)$ (note that we had used a coarse grained grid structure). Figure 3.4(b) shows such a workload-balanced micro-channel distribution. The grids with higher power density are allocated more channels, and within each grid $(i, j, \forall k)$, $N_{i,j}$ micro-channels spread uniformly if the TSVs do not block the placement of micro-channels.

To account for the presence of TSVs, during the micro-channel placement, when there are TSVs in any place along the micro-channel location, no micro-channel is allocated in this location.

3.3.4 Micro-channel Cost Assignment

Given the initial micro-channel distribution, we iteratively remove micro-channels to save pumping power as Algorithm 1 shows. To determine the order in which micro-channels are removed, we assign a cost to each micro-channel, which indicates the cost of removing this micro-channel. In each iteration, the micro-channel with the smallest cost is removed. After each micro-channel removal, the cost of remaining micro-channels is updated. In this subsection, we discuss how micro-

channel cost is assigned and updated.

Defining Micro-channel Cost: The temperature at an on chip location largely depends on the power dissipated in that region, and its neighboring regions. Thus, we use "weighted power" based approach for micro-channel cost assignment. Basically each micro-channel should absorb the heat generated in the region right below and above itself in active layers and also the heat generated in near neighbors. To assign the cost of micro-channels, we define a *region of influence (ROI)* for each potential micro-channel. The *ROI* of a micro-channel is the region to which this channel provides cooling (that is, the region right below and above this channel in active layer and also in the near neighbors). The dark region in Figure 3.6(a) shows the ROI of micro-channel 3. We divide the 3D-IC into fine grained grids, each of which contains at most one micro-channel. Let $W_{i,j}$ denote the cost of the micro-channel located in position (i, j) (i-th grid in x direction in micro-channel layer j), it is assigned as the weighted sum of the power dissipated in its ROI:

$$\begin{aligned} W_{i,j} =& u_1(w_0 \cdot P_{i,j+1} + \sum_{b=1}^{b_{max}} w_b \cdot (P_{i+b,j+1} + P_{i-b,j+1})) \\ &+u_2(w_0 \cdot P_{i,j-1} + \sum_{b=1}^{b_{max}} w_b \cdot (P_{i+b,j-1} + P_{i-b,j-1})) \end{aligned} \quad (3.7)$$

Here $P_{i,j} = \sum_{\forall k} P_{i,j,k}$, where $P_{i,j,k}$ is the power dissipated at grid (i, j, k) (the i-th/j-th/k-th grid in $x/y/z$ direction). In z direction the channel covers the whole chip, so we sum up the power in all grids in z direction (denoted by $P_{i,j}$) and the channel cost is a weighted sum of $P_{i,j}$.

The weight is decided by the distance from the heat source to the micro-

channel. In Equation 3.7, u_1 and u_2 are the vertical weight factors. Assume micro-channels absorb heat from the active layers right above and below them. As Figure 3.6(a) shows, u_1 is the vertical weight factor for the power from the active layer above the micro-channel, it is inversely proportional to the vertical distance between micro-channel and its top active layer. Similarly, u_2 is the vertical weight factor for the power from the active layer below the micro-channel, and its value is decided in a similar way. Here w_b is the horizontal weight factor. We assume horizontally each channel has a coverage of b_{max} in x direction, that is, each channel absorbs the heat in the region within a distance of b_{max} from it in x direction. Note that the horizontal distance here is measured in x direction since in z direction the channel covers the whole chip. The horizontal weight factor $w_b(b = 1...b_{max})$ is decided by the distance from the channel to the heat source in x direction (measured by b). We set $w_0 = 1$ and w_b is monotonically decreasing with distance b.

Updating Micro-channel Cost: After removing a micro-channel, we should update the cost of remaining channels. Basically, after a channel is removed, its neighboring channels should take care of the region covered by the removed channel (Figure 3.6(b)), and thus the cost of these neighboring channels should increase. Assuming (i_0, j) is the micro-channel we have just removed (the channel located at i_0-th grids in x direction in layer j), we will update the cost of remaining micro-channels in layers $j-2$, j and $j+2$ as Figure 3.6(b) shows (note that layers $j \pm 1$ are active layers), the update function is as follows:

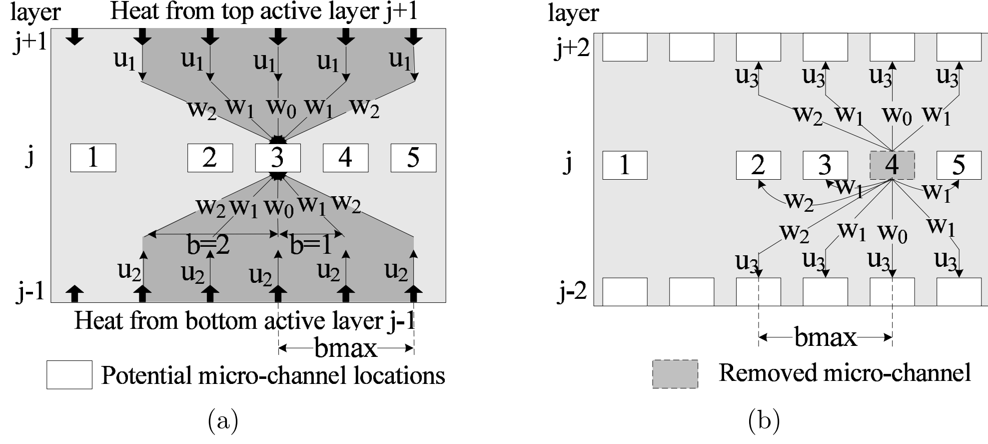


Figure 3.6: (a) Cost initialization, (b) Cost update

$$
\left.\begin{aligned}
W_{i,j} &= W_{i,j} + w_{|i_0-i|} \cdot W_{i_0,j} \\
W_{i,j\pm2} &= W_{i,j\pm2} + u_3 \cdot w_{|i_0-i|} \cdot W_{i_0,j}
\end{aligned}\right\} \forall i \; s.t. \; |i_0 - i| \le b_{max} \tag{3.8}
$$

Here w_b is the horizontal weight factor, and u_3 is the vertical weight factor decided by the vertical distance between two micro-channel layers.

The algorithm iterates until further removal of micro-channels results in thermal violation. The remaining micro-channels form the final cooling system. The cooling effectiveness of the resultant micro-channel design will be given in Section 3.7, which shows that the non-uniform micro-channel design can result in more than 50% pumping power savings compared with the conventional design.

Though significant power saving is achieved, this non-uniform micro-channel structure is still inefficient in dealing with the spatial constraints imposed by TSVs. In the next section, we will investigate a TSV constrained micro-channel design that can better address this problem and further save pumping power.

3.4 TSV Constrained Bended Micro-channel

3.4.1 Motivation of Using Bended Micro-channel

The previous configuration uses straight channels that spread in areas that demand high cooling. If the spatial distribution of micro-channels is unconstrained then such an approach results in the best cooling efficiency with the minimum cooling energy. However 3D-ICs impose significant constraints on how and where the micro-channels could be located due to the presence of TSVs, which allow different layers to communicate. A 3D-IC usually contains thousands of TSVs which are incorporated with clustered or distributed topologies [57]. These TSVs form obstacles to the micro-channels since the channels cannot be placed at the locations of TSVs. Therefore the presence of TSVs prevents distribution of straight micro-channels. This results in the following problems.

1. As illustrated in Figure 3.7(a), micro-channels would fail to reach thermally critical areas thereby resulting in thermal violations and hotspots.

2. To fix the thermal hotspots in areas where micro-channels cannot reach, we need to increase the fluid flow rate resulting in a significant increase in cooling energy.

To address this problem, we investigate micro-channel with bends as illustrated in Figure 3.7(b). With bended structure, the micro-channels can reach those TSV-blocked hotspot regions that straight micro-channels cannot reach. This results in better coverage of hotspots and therefore better cooling efficiency and reduced

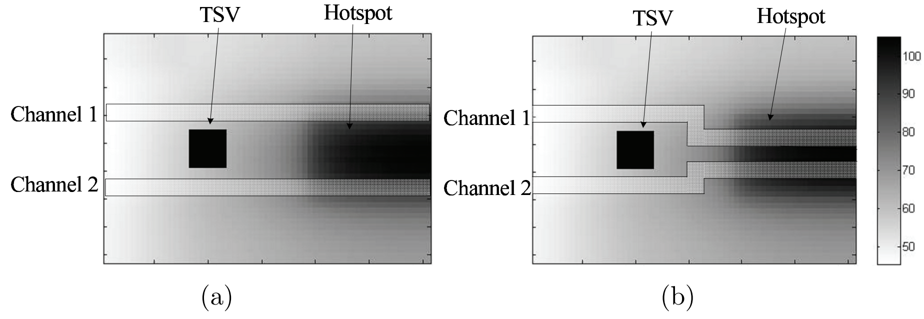


Figure 3.7: Example of silicon layer thermal profile with TSV and (a) straight, (b) bended micro-channels

cooling energy. While micro-channels with bends (or serpentine organization of micro-channels) have been investigated in the past [68][23], our work is the first one to investigate this structure from the context of 3D-ICs and more specifically address the constraint imposed by TSVs towards spreading of straight micro-channels [73].

3.4.2 Problem Formulation

In this work, we would like to decide the locations and geometry of micro-channels with bended structure so that its cooling effectiveness is maximized. Designing 3D-IC micro-channel infrastructure is a very complex problem. For example there are exponentially many ways to incorporate micro-channels with bends whose impact on the silicon temperature requires us to solve complex system of thermal equations. The specific problem formulation is as follows.

$$
\begin{aligned}
& min \quad P_{pump}(e^l_{i,j}, \Delta p) \\
& s.t. \sum_{\forall j \in \mathcal{N}(i)} e^l_{i,j} = 1, \forall \text{grid } i \in \{\text{CI, CO}\}, \forall \text{channel layer } l \\
& \sum_{\forall j \in \mathcal{N}(i)} e^l_{i,j} = k \in \{0, 2\}, \forall \text{grid } i \notin \{\text{CI, CO, TSV}\}, \forall \text{channel layer } l \\
& e^l_{i,j} = 0, \text{if grid } i \text{ or } j \in \{\text{TSV}\}, \forall \text{channel layer } l \\
& T^l_i(e^l_{i,j}, \Delta p) \leq T_{max}, \forall \text{grid } i, \forall \text{channel layer } l \\
& e^l_{i,j} \in \{0, 1\}, \forall \text{grids } i,\ j, \forall \text{channel layer } l \\
& e^l_{i,j} = e^l_{j,i}, \forall \text{grids } i,\ j, \forall \text{channel layer } l
\end{aligned}
\tag{3.9}
$$

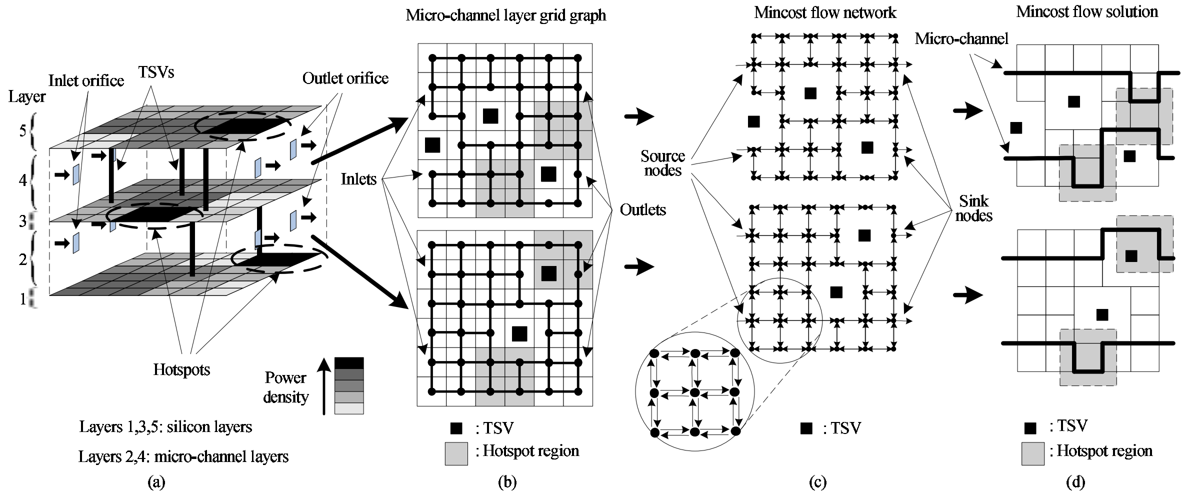


Figure 3.8: Example of micro-channel infrastructure design using minimum cost flow

Figure 3.8 represents the problem formulation graphically. Given a set of stacked silicon layers, some of the intermediate layers between silicon layers would have micro-channels (as shown in Figure 3.8(a), two intermediate layers comprise of micro-channels). The locations of input and output orifices for the micro-channels are assumed known. We would like to find micro-channel routes from one side to

the other such that the routes do not intersect, avoid TSVs and provide sufficient cooling at minimum pumping energy.

We impose a graph on each micro-channel layer as indicated in Figure 3.8(b). In the graph, each grid is represented by a node, and the edges define the immediate neighbors of a node. The micro-channel routing would be performed on this graph. If there is a TSV located on a grid, then its corresponding neighborhood edges are removed since micro-channels cannot be routed through TSVs. Let $e^l_{i,j} = 1$ represents the fact that there is a channel connecting grids i and j in the l-th micro-channel layer of the 3D-IC (so i and j must be neighboring nodes in the grid graph and $e^l_{i,j} = e^l_{j,i}$). Neither i nor j should have a TSV (because TSVs will not allow channels to go through them). In the first constraint, $\{CI, CO\}$ represents the set of input and output orifice nodes, $\mathcal{N}(i)$ represents the set of i's neighboring nodes. So the first constraint imposes that the input and output orifice nodes must have a neighboring grid they are connected to so that their incoming/outgoing fluid can be pushed into/out-of the micro-channel layer. The next constraint imposes that, for each grid, either there is a channel going through this grid (and therefore $\sum_{\forall j \in \mathcal{N}(i)} e^l_{i,j} = 2$), or no micro-channel goes through it (and therefore $\sum_{\forall j \in \mathcal{N}(i)} e^l_{i,j} = 0$). In the third constraint, $\{TSV\}$ represents the set of grids containing TSVs, so micro-channels cannot be routed through these nodes. The following constraint imposes that the temperature is within acceptable limits and the objective tries to minimize the pumping power.

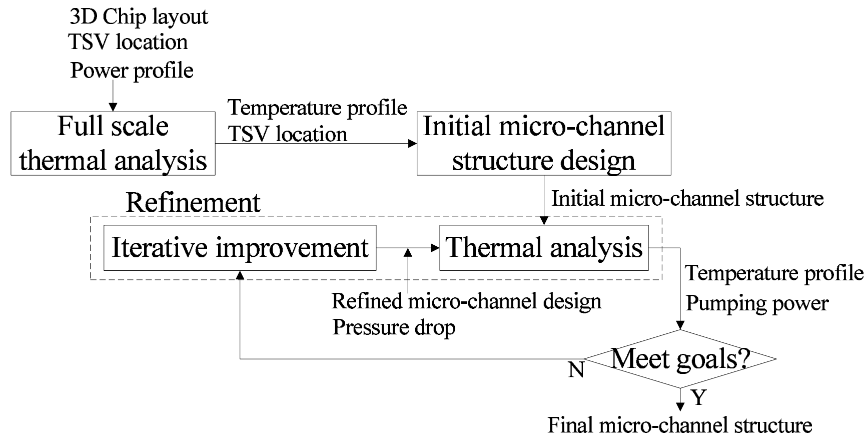


Figure 3.9: Micro-channel infrastructure design flow

3.4.3 Overall Micro-channel Design Flow

This is a very complex problem since: 1) the variables need to be discrete, and 2) the thermal and pumping power models are highly nonlinear. In this section we investigate such a methodology as illustrated in Figure 3.9. Our methodology follows a sequence of logical steps. First the severity of the thermal problem and the need for having micro-channels is evaluated by performing a full scale thermal analysis. Based on the severity of the thermal problem (location, intensity of hotspots) an initial micro-channel design is developed. This design is further improved for reducing the cooling power footprint and improving the thermal effectiveness using iterative methods. Now we go into the details of these individual steps.

3.4.4 Mincost Flow Based Micro-channel Design

The full scale 3D thermal analysis would identify locations of hotspots in different layers which cannot be removed by conventional package/air cooling based approaches. These are the areas which require sufficient proximity to the micro-

channels. Since solving the formulation in Equation 3.9 is intractable, we use simple models to come up with a *sufficiently good* initial micro-channel infrastructure which is iteratively improved subsequently. In order to develop this initial solution we use the minimum cost flow formulation.

3.4.4.1 Initialization of Minimum Cost Flow Network

Consider the 3D-IC and the corresponding grid graph of each micro-channel layer as illustrated in Figure 3.8(a)(b). For each micro-channel layer, we instantiate a minimum cost flow problem as follows (see Figure 3.8(c) for illustration). The nodes corresponding to the input/output orifices for the given micro-channel layer are assigned a supply/demand of one flow unit. All nodes in the grid graph have a capacity one. The edges have unlimited capacity and are bi-directional (can take fluid flow in either direction). As indicated earlier the edges between two neighboring nodes exist only if neither of the nodes has a TSV. This enforces the routing constraint imposed by TSVs. Figure 3.8(c) indicates the flow network for the two micro-channel layers.

Each node has a cost whose assignment would be discussed subsequently. We would like to send flow from inlet nodes to outlet nodes such that the capacity constraints are not violated and the cost is minimum. Assigning the node capacity to be 1 would ensure that all the flow from inlet to outlet follows simple paths (non-intersecting and non-cyclic). A minimum cost flow formulation with a well defined node capacity could be solved using very similar methods as a formulation with edge

capacity alone [65]. It is noteworthy that because there is an edge between each pair of neighboring nodes, the flow path could take several bends if necessary.

3.4.4.2 Cost Assignment

The cost assignment should be such that the minimum cost flow formulation develops an initial infrastructure that distributes the micro-channels with higher density in areas that demand more cooling. The chip scale thermal analysis would identify locations of grids in the silicon layers that are in dire need of cooling (see Figure 3.8(a)). A silicon layer would be cooled by the micro-channels both above and below (unless the silicon layer is at the very top or very bottom of the stack). For example, the middle silicon layer in Figure 3.8(a) could be cooled by two micro-channel layers unlike the top and bottom silicon layers.

As illustrated in Figure 3.8(b), each micro-channel layer is represented as a grid graph. The amount of cooling required at a certain node in this graph is a function of how hot the top and bottom grids in the silicon layers are. It also depends on how we chose to distribute the cooling demand at a certain location in the silicon layer between the micro-channel layers just above and just below. Let us suppose a certain location in the silicon layer has temperature $T \geq T_{max}$ and requires cooling (estimated by full scale thermal analysis). Let uT (with $0 \leq u \leq 1$) represent the fraction of this cooling demand assigned to the micro-channel grid right above and $(1-u)T$ represent the cooling demand assigned to the micro-channel grid just below. If u is set very low then most of the cooling will be done by the channel layer below

and vice versa for large u. Let u_i^l be the heat load partitioning factor of grid i in silicon layer l, it is assigned as follows.

Case 1: If l is the topmost (bottommost) layer, then $u_i^l = 0 (u_i^l = 1)$ so that all the cooling demand goes to the micro-channel layer right below (above) l, which is layer $l-1$ $(l+1)$.

Case 2: If l is neither top nor bottom layer, $0 \leq u_i^l \leq 1$, implying that the heat generated in grid i of silicon layer l needs to be distributed in the two micro-channels layers right above and below. If the channel layers above and below (layers $l+1$ and $l-1$) have the same number of TSVs then $u_i^l = 1/2$, else it is scaled linearly such that more cooling demand is assigned to the micro-channel layer with lesser TSVs.

Given the partitioning factor u_i^l, the cost is assigned as follows. (See Figure 3.10 for an illustration.) Let $cost(i, l)$ denote the cost for node i in micro-channel layer l (hence layers $l-1$ and $l+1$ correspond to silicon layers just below and above the micro-channel layer l), three cases are considered depending on whether there is hotspot below and above this node in the silicon layers $l-1$ and $l+1$.

Case 1: Hotspots on both sides. When the grid i in both silicon layers $l-1$ and $l+1$ are in hotspot regions ($T_i^{l-1} > T_{max}$ and $T_i^{l+1} > T_{max}$), the micro-channel should provide cooling to both sides (above and below), so the cost is:

$$cost(i, l) = -[(1 - u_i^{l+1})T_i^{l+1} + u_i^{l-1}T_i^{l-1}] \tag{3.10}$$

Here the first component inside the square bracket indicates the cooling demand from the silicon grid above and the second component corresponds to the cooling

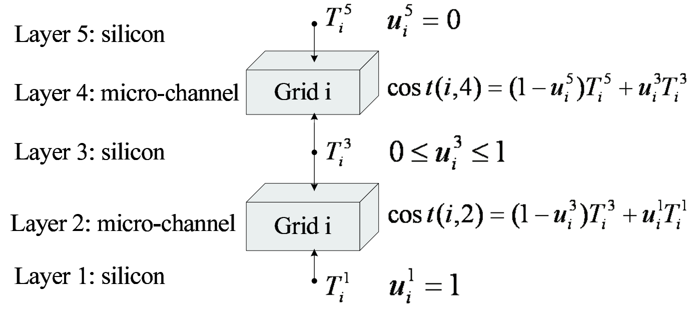


Figure 3.10: Cost assignment

demand from the silicon grid just below. Higher demand leads to lower cost since we would like micro-channels to pass through high cooling demand regions. See Figure 3.10 for an illustration.

Case 2: Hotspot in one side. When the silicon grid i on only one side ($l-1$ or $l+1$) is in hotspot region (but not both), the cost is assigned as

$$cost(i,l) = \begin{cases} -(1-u_i^{l+1})T_i^{l+1}, \text{ if } T_i^{l+1} \ge T_{max} \\ -u_i^{l-1}T_i^{l-1}, \text{ if } T_i^{l-1} \ge T_{max} \end{cases} \tag{3.11}$$

Case 3: No hotspot in either side. When there is no hotspot in either side, then the node cost is assigned to a small positive value $cost(i,l) = \epsilon > 0$.

The minimum cost flow formulation would therefore route flows such that maximum number of high cooling demand grids are touched by the channels. The non-hotspot regions are assigned a small positive cost. This would enable the minimum cost flow formulation to avoid areas that do not demand high cooling.

3.4.5 Micro-channel Refinement

The primary objective of the minimum cost flow formulation is to come up with an initial micro-channel design that carries cooling in sufficient proximity of hot areas. This is not enough to guarantee effective cooling. For example, some channels have several bends and/or may be routed over disproportionately large number of hotspots. Both of these situations cause a degradation in the overall cooling quality. In this section we present approaches for iteratively refining the design for improved cooling effectiveness. The micro-channel infrastructure refinement process works as illustrated in Figure 3.9.

3.4.5.1 Temperature and Pumping Power Analysis

The impact of micro-channels on the 3D-IC thermal profile is a function of how the micro-channels are routed and also how much fluid flow they carry. The initial design generated using minimum cost flow technique does not prescribe the pressure drop and the fluid flow rate that the channels need to work at. Hence given the micro-channel design, we then need to estimate the *smallest* pressure drop that the pump needs to work at such that thermal constraints are satisfied. Given the micro-channel design, the smallest pressure drop value results in the smallest pumping energy. As indicated earlier, we assume that all channels are subjected to the same pressure drop by the pump, hence the minimum pressure drop can be determined by linearly increasing pressure drop (Δp) and calculating the thermal profile for each value until the thermal constraints are met. For a given pressure

drop across the pump and a given micro-channel design, Equation 2.21 could be used to determine the velocity (fluid flow rate) in each channel. Note that because each channel has different number of bends and total length, the flow rate would be different too. Based on the flow rate information which is computed for a given pressure drop, the associated thermal conductance matrix **G** could be computed. This information could be used to estimate the thermal profile of the 3D-IC for a given pressure drop. After finding the minimum required pressure drop (Δp), we could calculate the required pumping power. This technique is highlighted in Algorithm 2.

Algorithm 2 Finding the minimum required pumping power

1. $\Delta p = \Delta p_{min}$, and repeat steps 2-6:
2. Calculate the fluid velocity using Equations 2.21;
3. Calculate thermal conductance matrix **G**;
4. Estimate temperature profile;
5. If thermal violation occurs, $\Delta p = \Delta p + \delta p$;
6. Else break;
7. Calculate pumping power.

3.4.5.2 Iterative Micro-channel Optimization

The objective of minimum cost flow formulation did not capture cooling energy and/or number of bends in the channels. Figure 3.11 illustrates typical situations that can occur. In Figure 3.11, the two micro-channels have significantly different cooling demands (Figure 3.11(a)) and number of bends (Figure 3.11(b)). Such imbalance (in cooling demand and bend count) leads to increase in the required pressure drop and thereby increasing the pumping energy. The basic idea is that all the channels should have similar levels of heat load, length and number of bends.

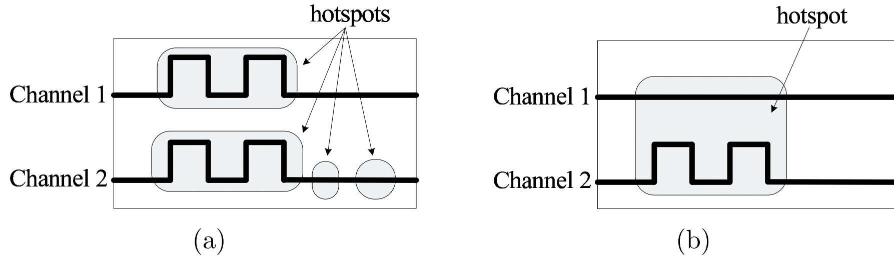


Figure 3.11: Examples of (a) unbalanced cooling demand, (b) different number of bends

Hence if a channel has too many bends or goes through many hotspots while others are shorter, then other channels could be made longer thereby more uniformly distributing the heat load and also reducing the number of bends in the most *critical* micro-channel.

Based on these considerations, we try to refine the initial design by 1) balancing the heat loads among micro-channels and 2)reducing unnecessary bends.

Micro-channel heat load balancing:

Starting from the initial design we identify the micro-channels which have disproportionately high heat removal load and spread their heat load into neighboring channels.

Algorithm 3 highlights the iterative pairwise micro-channel cooling load balance process. In the first iteration of pairwise micro-channel cooling workload balance, we start from the channel with the highest cooling workload. Here the cooling workload is measured by the total heat absorbed by the micro-channel, which could be calculated using $P = (T_{out} - T_{in})/R_{io}$. Here T_{in} is the fluid supply temperature at micro-channel inlet, and T_{out} is the fluid temperature at micro-channel outlet, R_{io} is the total thermal resistance between the fluid inlet and outlet of that spe-

cific channel. Given the pressure drop, power profile of the 3D-IC and the location and dimensions of the micro-channels, these parameters could be easily calculated (see discussion in Sections 2.3 and 2.4, as well as reference [76]). Assuming i is the channel with the highest cooling workload, we then pick one of i's neighbors (either left or right) with lower cooling workload, say channel k, and balance the workload between channels i and k.

Algorithm 3 Pairwise micro-channel cooling load balance

Repeat:
1. Pick the micro-channel with highest cooling load i;
2. Pick a micro-channel k from i's neighbor with smaller cooling load, that is, $k = argmin_{k \in \{i-1, i+1\}}(load(k))$;
3. Equally divide the hotspot region covered by channels i and k, and assign one of the region to channel i, the other to channel k;
4. Remove some edges on the boundary between these two regions from the grid graph;
5. Resolve the minimum cost flow based on new graph;
6. Temperature analysis and calculating minimum required pumping power using Algorithm 2;
7. If no further pumping power saving could be achieved, stop.

To balance the workload of channels i and k, we firstly partition the hotspot regions covered by channels i and k. This region is bounded by channels $min(i, k)-1$ and $max(i, k) + 1$. For instance, as shown in Figure 3.12 in which we would like to balance the workload between channels 2 and 3. Then, the hotspot region covered by channels 2 and 3 is bounded by channels 1 and 4 (region identified by dotted line in Figure 3.12). To equally partition this region, basically, we would like the resultant two parts have similar total amount of heat load (cooling demand). As indicated earlier, the cost of a node i at the l-th micro-channel layer signifies the degree of cooling desired there. The total cooling needed in the region covered by channels i and k is simply the sum total of the cost in all the associated grids. We

would like each channel to be assigned about half of this total cooling load in that region. Hence we would like to partition this region into two subregions with the same total cooling load.

Starting from the top left grid of the region covered by i and k, we traverse the grid network in a row major form (left to right and then bottom). As soon as we have collected grids whose sum total of cooling load is 1/2 of that of the region, we stop. The boundary between these two subregions is defined in this fashion. A row major form of traversal ensures that each channel will be somewhat uniformly loaded with heat from a spatial perspective. Now one region is assigned to i and the other is assigned to k. In order to find the exact route of the micro-channels we can remove the edges connecting the two regions and solve the minimum cost flow formulation once again (see Figure 3.12). This would ensure that channels i and k do not encroach on each others regions. In the case where the minimum cost flow could not return feasible solution due to the removal of too many edges, we will add some removed edges back until a feasible solution is returned.

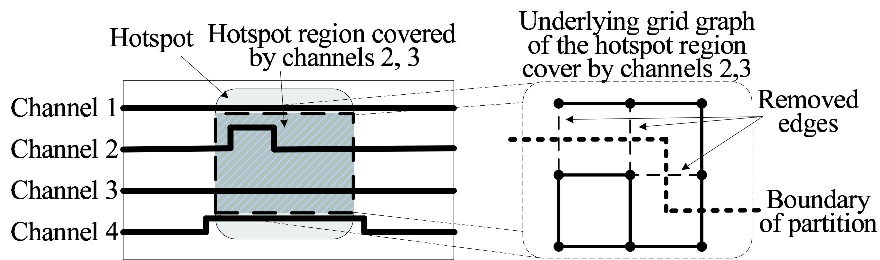


Figure 3.12: Example of pairwise cooling workload balance

The minimum cost flow gives a refined micro-channel structure design. We then redo the temperature analysis and find the minimum pumping power for the new design using algorithm 2.

In the next iteration of optimization, we find the currently highest workload micro-channel in the new design and do pairwise load balance on this channel using the new graph updated in the previous iteration. We repeat this process iteratively until no further pumping power saving could be achieved.

Bend Elimination

As shown in section 2.4.3.2, the corners/bends in the micro-channel will introduce considerable pressure drop, which increases the pumping power. Bends in micro-channels allow us to reach areas which cannot be directly connected due to the presence of TSV obstacles. But unnecessary bends which have been incorporated due to the heuristic nature of our algorithm provide little benefit while impacting the cooling quality. As a final refinement step we develop a pattern matching based scheme for removing unnecessary and redundant bends on the channel networks.

We firstly generate a library of the patterns of unnecessary corners and use pattern match to find those unnecessary corners in our design. Then, we replace those corner patterns with some equivalent patterns with lesser corners. Figure 3.13 highlights a few patterns and their replacement patterns. This step should be performed in a judicious fashion. Removing corners in the hotspot region might lead to reduction in the micro-channel cooling performance since it reduces the level of coverage. Hence we only remove those corners in the non-hotspot regions which can easily be identified by the thermal analysis. The algorithms used for pattern matching are similar to those used in technology mapping. The exact details of how pattern matching is done has been omitted here.

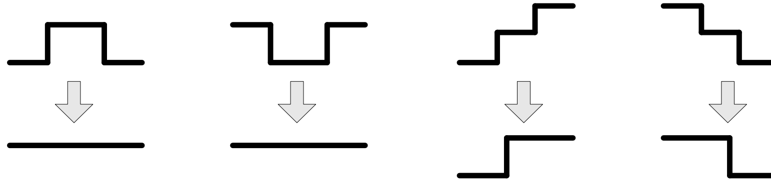


Figure 3.13: Examples of bend elimination

3.5 Hybrid Cooling Network

3.5.1 Motivation of Hybrid Cooling Network

Besides micro-channels, TSVs are also considered as an alternative solution for cooling of 3D-ICs. TSVs are usually made of copper which has better thermal conductivity than silicon or metal-oxide, and hence enable better vertical heat conduction between different layers. When the number of signal TSVs is not enough, dummy thermal TSVs are inserted to further mitigate the thermal issues.

Both micro-channels and thermal TSVs have advantages and drawbacks in performing 3D-IC cooling.

Micro-channel liquid cooling: The cooling effectiveness of micro-channel is quite high and they have been reported to support heat densities as high as $700W/cm^2$ [84]. However as illustrated earlier, the drawback of micro-channel based heat removal technology is that the cooling system consumes extra energy for pumping the coolant through channels. On the other hand, the presence of TSVs that connect signals and power between layers constraints the locations where channels could be placed, since micro-channels cannot be placed in the locations where these TSVs are allocated (as shown in Figure 3.1). This constraint limits the heat removal

capability of micro-channels.

Thermal TSV: The thermal TSVs help alleviate the 3D-IC thermal issues by establishing heat transfer paths from heat source to heat sink using high thermal conductivity materials, so that heat can be more effectively absorbed by heat sinks. It also moves heat from hot to cool areas (without consuming extra cooling power) to balance the heat between layers and make the thermal profile more uniform. However, thermal TSVs only help redistribute heat instead of removing heat. Moreover, since the TSVs can only be placed in the whitespace between the layout, the number and locations of thermal TSVs are limited by the chip floorplan. As a result, their cooling capability is limited. Also, large number of TSVs will increase the fabrication cost, degrade the yield of chips and exacerbate the thermal stress problem in 3D-IC.

Based on these considerations, in this section, we propose a hybrid 3D-IC cooling scheme: a cooling network which uses micro-channel based liquid cooling together with thermal TSVs [75]. In this hybrid cooling network, micro-channels and thermal TSVs work in a mutually complementary way. Thermal TSVs redistribute heat and establish heat dissipation paths that deliver heat to micro-channels, and the heat is then removed by micro-channels. This hybrid cooling scheme would provide sufficient level of cooling to the 3D-IC using fewer cooling power and thermal TSVs. To extract maximum cooling effectiveness, we would like to co-optimize the allocation of micro-channels and thermal TSVs.

3.5.2 Algorithm for Hybrid Cooling Network Design

Our algorithm for micro-channel and thermal TSV co-optimization is based on iterative improvement. The overall iterative design flow is similar as the algorithm in Section 3.3. But instead of iteratively removing micro-channels, we use a constructive approach. That is, we start from the 3D-IC structure **without** any micro-channel or thermal TSV, and iteratively **add** micro-channels and size thermal TSVs until they could provide sufficient cooling. The overall constructive design approach is illustrated in Algorithm 4 and Figure 3.14.

Algorithm 4 Heuristic for micro-channel and thermal TSV co-optimization

Starting from the 3D-IC structure without micro-channels or thermal TSVs:
1. Assuming we are given a set of potential micro-channel locations, initialize the priority level of each potential micro-channel;
2. Repeat until thermal constraint is satisfied:
3. Add a micro-channel with highest priority;
4. Decide the locations and sizes of thermal TSVs;
5. Set up thermal resistive network, estimate thermal profile;
6. If thermal constraint is satisfied, stop;
7. Else update priority of un-added channels and go to step 2.

The algorithm starts by finding a set of potential locations for micro-channels. We use the algorithm proposed in Section 3.3.3 to find the set of potential micro-channel locations.

Based on the potential micro-channel locations, we assign a priority for each potential micro-channel. The priority is associated with the significance of the micro-channel in removing heat. In each iteration, we firstly add a micro-channel with the highest priority (that is, the most important micro-channel). Then we insert or size thermal TSVs based on the current micro-channel allocation. After the micro-channel and thermal TSV placement in each iteration, we check if the current cooling

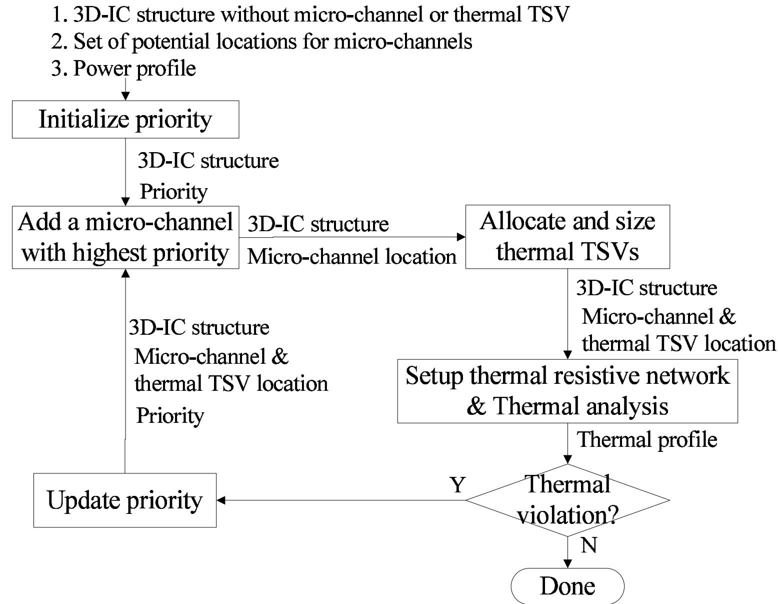


Figure 3.14: Overall design flow of micro-channel and thermal TSV co-optimization

system design could provide enough cooling to the 3D-IC. If not, we will continue to add more micro-channels and resize thermal TSVs. Once we have added a micro-channel, we need to update the priority of the remaining un-added micro-channels before adding another micro-channel. We repeat this iterative process until thermal constraint is satisfied.

The success of this approach depends on how micro-channel priority is assigned and how thermal TSVs are allocated and sized. The next three subsections explain them in detail.

3.5.3 Micro-channel Priority Assignment/Update

The micro-channel priority assignment and update is similar as the micro-channel cost assignment/update approach presented in Section 3.3.4, with slight modifications on the updating formulation as Equation 3.12 shows.

$$\left.\begin{aligned} W_{i,j} &= W_{i,j} - w_{|i_0-i|} \cdot W_{i_0,j} \\ W_{i,j\pm2} &= W_{i,j\pm2} - u_3 \cdot w_{|i_0-i|} \cdot W_{i_0,j} \end{aligned}\right\} \forall i \ s.t. \ |i_0 - i| \leq b_{max} \tag{3.12}$$

Basically, when we add a micro-channel, this micro-channel absorbs heat from the regions surrounding it, so the cooling workload of its potential neighboring micro-channels would reduce. Hence the priority of the potential neighboring micro-channels should decrease as Equation 3.12 shows.

3.5.4 Thermal TSV Allocation and Sizing

After inserting a micro-channel in each iteration, we place thermal TSVs in the remaining available area to further reduce the chip temperature. For thermal TSV allocation and sizing, we use the basic idea of iterative thermal conductivity updating proposed in [24], but improve it for better rate of convergence.

3.5.4.1 Basic Thermal TSV Placement Approach

In the approach proposed in [24], the 3D-IC is divided into fine grids. It finds the distribution of thermal TSVs by calculating the desired vertical thermal conductivity of each grid that could eliminate or mitigate thermal problem. Their approach is based on iterative improvement. To update the thermal conductivity in each iteration, the vertical thermal gradient q_z between two vertically neighboring grids is calculated, and the vertical thermal conductivity k_z in each grid is updated using the following equation:

$$k_z^{new} = \frac{q_z^{old}}{q_z^{new}} k_z^{old} \tag{3.13}$$

where q_z^{old} is the current vertical thermal gradient, and the new thermal gradient q_z^{new} (which is the desired thermal gradient after this iteration) is chosen as some value closer to the ideal thermal gradient q_{ideal} than q_z^{old}:

$$|q_z^{new}| = q_{ideal} (\frac{|q_z^{old}|}{q_{ideal}})^{\theta} \tag{3.14}$$

Here θ is a user defined parameter between 0 and 1, which is used to control the rate of convergence. In each iteration, the thermal conductivity of all grids is updated simultaneously. Once the algorithm converges, they calculate the number/size of thermal TSVs in each grid that could result in the desired thermal conductivity using Equation 2.9.

Adding a thermal TSV will change the thermal conductivity matrix **G** (given in Section 2.3.1) and hence change the thermal gradient q_z across the chip. So basically every time we have placed or sized a thermal TSV, we need to recompute the thermal profile and get the updated thermal gradient q_z before updating the thermal conductivity of the next grid. Nevertheless, in [24], the thermal conductivities of all grids are updated simultaneously in each iteration. In order to simultaneously update the thermal conductivity of all grids without recalculating the thermal profile, the parameter θ should be close to 1 so that the change in thermal conductivity in each step is very small and therefore has little influence on the thermal gradient of other grids. However using such a θ value leads to slower convergence rate.

3.5.4.2 Modified Thermal TSV Allocation and Sizing Approach

In our modified thermal TSV planning approach, we still use the basic iterative updating framework given in [24]. However, as explained earlier, the approach proposed in [24] needs to use a large θ which indicates slower rate of convergence. So in our modified approach, instead of modifying the thermal conductivity in all grids in each iteration, we only update a subset of the grids $\mathcal{E}$. The grids in this subset $\mathcal{E}$ should satisfy the following two conditions: a) all the grids in this set have very small interdependence with each other, and b) they have large influence on the hotspot regions. The first condition ensures that only those grids that are independent of each other are updated. So when we change the thermal conductivity of a grid in set $\mathcal{E}$, the thermal gradient of other grids in this set almost does not change. Hence we could simultaneously update all the grids in this set using a small θ which indicates faster rate of convergence. The second condition ensures that we focus on updating those grids that are most likely to reduce the hotspot temperature. This could help us to reduce the number and size of thermal TSVs used. We call this subset "*maximum independent set* $\mathcal{E}$".

The success of this approach depends on how many independent grids we could find and simultaneously update without recomputing the thermal profile in each iteration. The micro-channel heat sinks basically behave as heat isolators (since they carry heat away) and therefore reduce the interdependence between grids. Hence the existence of micro-channels leads to more independent grids that can be updated simultaneously.

Based on these two conditions, our modified thermal TSV placement and sizing algorithm works as follows:

Algorithm 5 Algorithm of thermal TSV placement and sizing

1. Estimate interdependency of each pair of grids;
2. Repeat steps 3-6 until the stop condition is satisfied:
3. Assign a weight to each grid according to its interdependency with hotspot grids;
4. Find the maximum independent set $\mathcal{E}$;
5. Update the thermal conductance of the grids in set $\mathcal{E}$ using the approach given in Section 3.5.4.1;
6. Update thermal gradient and grid interdependency, go to step 2.
7. Calculate thermal TSV size/density in each grid based on the achieved thermal conductivity.

In the next subsection, we explain how to find the maximum independent set $\mathcal{E}$ in detail.

3.5.4.3 Finding Maximum Independent Set $\mathcal{E}$

For a given 3D-IC structure, to estimate the interdependency between grids, we firstly calculate the inverse of thermal conductance matrix $\mathbf{G}$. This inverse matrix $\mathbf{H} = \mathbf{G}^{-1}$ satisfies $T = \mathbf{H} \cdot Q$. Here $H(i, j)$ basically indicates how much temperature increase in grid i is caused by the power dissipation in grid j. If $H(i, j) > 0$, when the thermal conductivity at grid j changes, it will affect the temperature at grid i.

The interdependency of each pair of grids depends on how many power sources they share. Here we use interdependency matrix INT to indicate the interdependency between each pair of grids. The interdependency matrix is defined as:

$$INT(i, j) = INT(j, i) = \begin{cases} 1, & \text{if } H(i) \cdot H^T(j) > \zeta \\ 0, & \text{otherwise} \end{cases} \tag{3.15}$$

Here INT is symmetric. $INT(i,j)$ indicates whether grids i and j are interdependent (1 indicate the two grids are interdependent and 0 indicate they are independent). $INT(i,j)$ is decided by the correlation between grids i and j which is measured by $H(i) \cdot H^T(j)$ ($H(i)$ represents the i-th row of matrix $\mathbf{H}$). When the correlation is very small (less than ζ), we assume the two grids are independent and set $INT(i,j)$ to 0, otherwise, we set it to 1 which indicates the two grids are dependent.

Once we get the interdependency matrix, we would like to find the set of grids that are: a) independent of each other and b) have maximum dependency with hotspot grids. To achieve this, we assign a weight to each grid which indicates its interdependency with all hotspot grids, and then find the set of independent grids with the maximum total weights.

Grid weight assignment

The weight of each grid c_i which represents its interdependency with hotspot regions is assigned as follows:

$$c_i = INT(i) \cdot \vec{E}^T, \text{for each grid } i \tag{3.16}$$

where $INT(i)$ is the i-th row of interdependency matrix INT, and $\vec{E} = \{E_j, \forall grid\, j\}$ is a vector indicating whether each grid is a hotspot:

$$E(j) = \begin{cases} 1, \text{ if } T_j > T_{max} \text{ where } T_j \text{ is temperature of grid } j \\ \\ 0, \text{ otherwise} \end{cases} \tag{3.17}$$

Here T_{max} is the thermal constraint.

If the weight of a grid is high, this basically means that the grid has higher interdependency with hotspots. We would like to focus on updating those grids that have higher interdependency with hotspots since inserting thermal TSVs in these grids can better reduce the hotspot temperature.

Finding independent grids with maximum total weight

Given the weight of each grid and the interdependence between them, we would like to find the set of grids which are independent and have the maximum total weight. This problem is equivalent to weighted clique problem which is NP complete. Many existing works have proposed heuristics to find a good solution. Here we use the adaptive, randomized greedy approach in [31].

Once we get this maximum independent set $\mathcal{E}$, we simultaneously update the thermal conductivity of the grids in this set using the approach illustrated in Section 3.5.4.1. Since the grids in this set have very small interdependence with each other, we can use a θ close to 0 thereby achieving faster convergence rate. Moreover, because we only update grids that are highly interdependent with hotspots, we could use fewer thermal TSVs.

Updating interdependence matrix

The change in thermal TSVs will change the thermal resistive network thereby changing the grid interdependency. So after we updated the thermal TSV in each iteration, we should update the interdependence matrix based on the new thermal resistive network. A simple approach is to regenerate the thermal conductance matrix $\mathbf{G}$ and then recalculate its inverse matrix $\mathbf{H}$ as well as the interdependency matrix INT after every iteration. However, the problem is calculating the inverse matrix $\mathbf{H}$ is time consuming. To save time for computing interdependence matrix, we only calculate (initialize) matrices $\mathbf{H}$ and INT once at the beginning of the algorithm before allocating or sizing any thermal TSV, and every time we updated thermal TSV, we only update some elements of matrix INT instead of re-calculating the whole matrix.

By exploring the interdependency matrix, we found that, the interdependency between two grids largely depends on the distance between them. We define an *interdependence region* for each grid, which includes all the grids that are interdependent with this grid. We found that, each grid usually has higher interdependence with the grids close to it and smaller or no interdependence with those grids far away. So the interdependence region of a grid is usually a region surrounding that grid as Figure 3.15 shows.

As we have added or enlarged a thermal TSV, the interdependency between grids would generally increase because the thermal conductivity increased. So the interdependence region of each grid is enlarged (as Figure 3.15(a) shows). On the other hand, as we reduce the size of a thermal TSV, the interdependency between grids reduces and the interdependence region of each grid shrinks (as Figure 3.15(b)

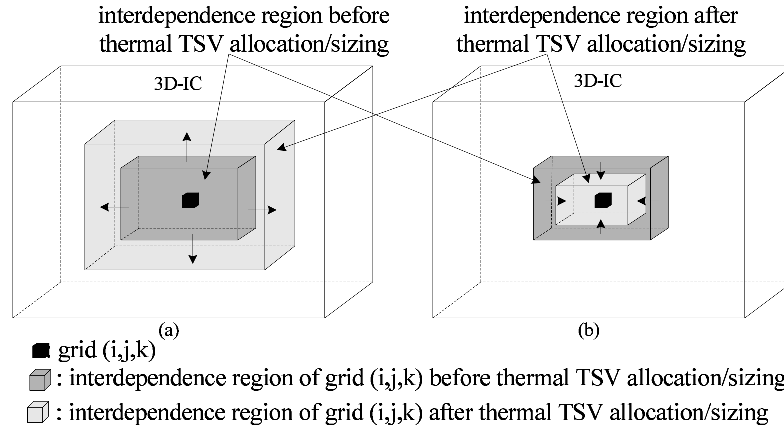


Figure 3.15: Change in interdependence region of a grid (a) after allocating or enlarging a thermal TSV, (b) after shrinking a thermal TSV

shows). Usually the change in a thermal TSV only affects the interdependence regions of the grids close to this TSV. The level by which we enlarge/shrink the interdependence region of each grid depends on the distance between this grid and the newly allocated/sized thermal TSV, and also depends on the amount by which we have sized the thermal TSV.

Once we updated the interdependence region of the grids, we can modify the interdependence matrix INT based on the new interdependence region of each grid.

Stop condition: We keep updating the thermal conductivities iteratively until one of the following situations occurs:

a) Thermal constraint is satisfied. In this case, no more thermal TSV is needed.

b) The thermal TSV capacity is reached. In this case, no more thermal TSV could be added.

c) Peak temperature cannot be further reduced. In this case, the algorithm converges, so adding more thermal TSV will not help reducing the chip temperature.

After the thermal TSV allocation/sizing, we perform thermal analysis and

check if the resultant micro-channel and thermal TSV allocation could provide enough cooling to the 3D-IC. If the resultant maximum temperature is within the thermal constraint, then the current design is our final design. Otherwise, we will continue to add micro-channels and size thermal TSVs until thermal constraint is satisfied.

3.6 Considering Thermal Variations

The previous approaches in Sections 3.3-3.5 assume that the power profile is fixed and known, and design the cooling structure based on the given power profile. In reality CPU power profiles are strong function of the application and vary based on the workload the CPU is experiencing at a given time. We address this problem by using multiple training power profiles. Given a set of training power profiles (that represent different classes of applications and workload levels), we would design the cooling structure (non-uniform, bended micro-channel or hybrid cooling system) which provides enough cooling to all the power profiles using minimum amount of pumping power. Conventionally such approaches are addressed by choosing the profile with the highest total dissipated power (TDP) and designing the cooling system based on it. But such approach fails to account for the fact that a power profile with a smaller TDP might end up with thermal violations due to the nature of its hotspots even if the profile with higher TDP does not. The advantage of using multiple training power profiles is that the resultant micro-channel network could adapt to various power profiles.

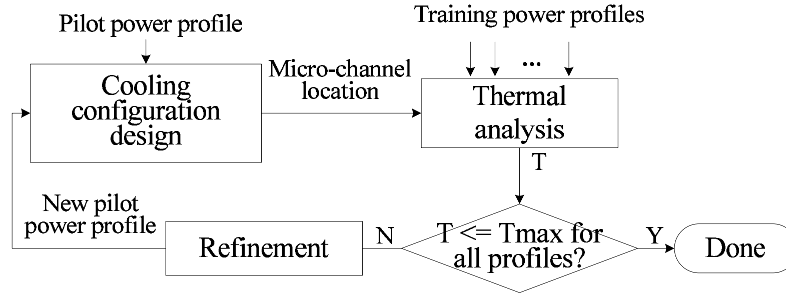


Figure 3.16: Flow chart of micro-channel placement

Our approach that accounts for multiple power profiles is illustrated in Figure 4.3. We start with the power profile with the highest TDP and design the cooling structure for this power profile using the heuristics given in Sections 3.3-3.5. We call this a pilot power profile. Then we test if all the power profiles meet the thermal constraint. If a set of power profiles violate the thermal constraint, then the pilot power profile is refined using Algorithm 6 and the cooling structure is re-designed based on the new pilot power profile.

Algorithm 6 Pilot power profile refinement

Assuming temperature constraint violation occurs in power profiles $\vec{P^1}, \vec{P^2}, ..., \vec{P^M}$;
1. For m = 1 to M
2. Increase power density of pilot power profile in the region where thermal violations occurs in power profile $\vec{P^m}$.

The refining process in step 2 of Algorithm 6 is basically increasing power density of pilot power profile in the regions where thermal violation occurs in the other power profiles. This would enable the micro-channel placement heuristic to allocate more cooling (either micro-channels or thermal TSVs) in that region. For example, if the violation occurs at grid (i, j, k) of power profile $\vec{P^m}$, we increase the power consumption at grid (i, j, k) and all grids surrounding (i, j, k) in the pilot power profile. The level of increase depends on the degree of thermal violation and

the distance from (i, j, k). The performance of this heuristic depends on the range in which we choose to increase the power in the pilot profile. If this range is large, the algorithm will converge faster but might have more channels and therefore higher pumping power.

3.7 Cooling Performance of Micro-channel Designs

Now we compare the cooling effectiveness of the three micro-channel designs. In our experiment, we use a three-tier stacked 3D structure. In the 3D-IC, three active layers are vertically stacked and the micro-channel layers are below each active layer. There is also an air-cooled heat sink at the top of 3D-IC. We use the ITC'99 circuits, which are typical synthesized circuits consisting of AND, OR, NOT, NAND and NOR gates, to generate the 3D-IC benchmarks [4]. Each 3D-IC layer contains several arbitrarily chosen ITC'99 circuits. We use the Capo placer to place the gates in each layer [1]. To obtain the power profiles for each layer, we randomly assign a switching activity factor (between $[0, 1]$) for each gate and use the power models in [47][86] to estimate the power consumption. Based on the placement information, we also find the whitespace between layout, and randomly allocate 1000 signal TSVs in the whitespace. This forms our testing benchmarks.

The chip dimension is $W = L = 9mm$. We setup the resistive network by using the hotspot like model in three dimension [77]. The micro-channel width and height is $\Delta x = 100\mu m$ and $\Delta z = 200\mu m$, and the diameter of TSV is $10\mu m$. The overall thermal resistance of the heat sink for air cooling is 0.5℃/W. The inlet

coolant temperature is 10°C and the maximum temperature constraint T_{max} is 85°C.

We compare the pumping power of the three micro-channel designs proposed in this chapter. The comparison is given in Table 3.1 and Figure 3.17. For all the power profiles, the air cooling cannot provide sufficient cooling to reduce the chip temperature below thermal constraint. Here, *All channels* design indicates the conventional micro-channel design that spreads straight micro-channels all over the interlayer regions, and *save* indicate the pumping power saving of each design over the *All channels* design. As we can see from the table, the *Non-uniform* micro-channel design saves about 57% pumping power compared with the *All channels* design. Using bended micro-channel could save another 11% pumping power. Among these three approaches, the *Hybrid* cooling network saves most pumping power (78% pumping power savings compared with the conventional *All channels* design).

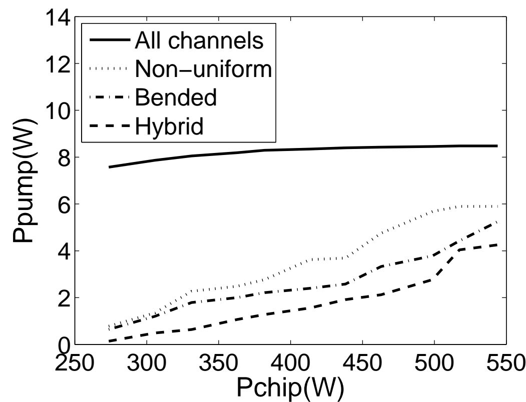


Figure 3.17: Comparison of Pumping Power

3.8 Runtime Thermal Management Using Micro-channels

Recently, the micro-fluidic cooling has also been adopted in dynamic thermal management (DTM) to control the runtime CPU performance and chip temperature

Table 3.1: Comparison of pumping power

	All channels		Non-uniform			Bended			Hybrid		
P_{chip}	N	P_{pump}	N	P_{pump}	save	N	P_{pump}	save	N	P_{pump}	save
273.6	90	7.6	9	0.7	91%	8	0.6	92%	2	0.1	99%
305.7	90	7.9	16	1.3	84%	14	1.2	85%	6	0.5	94%
331.2	90	8.1	25	2.2	73%	20	1.7	79%	7	0.6	93%
362.7	90	8.2	27	2.4	71%	22	2.0	76%	12	1.0	88%
381.9	90	8.3	30	2.7	67%	24	2.2	73%	14	1.2	86%
413.5	90	8.3	39	3.6	57%	26	2.4	71%	17	1.5	82%
438.1	90	8.4	39	3.7	56%	28	2.5	70%	20	1.9	77%
462.9	90	8.4	51	4.7	44%	36	3.3	61%	23	2.1	75%
498.7	90	8.5	60	5.6	34%	40	3.7	56%	29	2.7	68%
517.3	90	8.5	62	5.9	31%	47	4.4	48%	43	4.0	53%
544.1	90	8.5	62	5.9	31%	56	5.2	39%	45	4.2	51%
Average	90	8.2	38	3.5	57%	29	2.6	68%	20	1.8	78%

by tuning the fluid flow rate through micro-channels [19][18][61].

In this section, we investigate a micro-channel based DTM scheme that could provide sufficient cooling to the 3D-IC using minimal amount of cooling energy [71]. In this DTM scheme, assuming the micro-channel structure has already been decided using either of the aforementioned structures (Sections 3.3-3.5), it dynamically controls the pressure drop across the micro-channels based on the runtime cooling demand. Now we explain our micro-channel based DTM scheme in detail.

3.8.1 Algorithm for Micro-fluidic Based DTM

The temperature profiles on chip is a strong function of the power dissipated, while the power dissipation depends on the applications which change at runtime.

In order to track the runtime thermal and power state, thermal sensors are placed at various chip locations. Our micro-channel based DTM keeps track of power profiles at runtime using the information achieved by thermal sensors and adaptive Kalman filter based estimation approach (proposed in [96]), and then decides the micro-channel pressure drop based on it.

To estimate the power profile, [96] assumes there are M different power states (power profiles), each of which essentially represents a certain class of applications. The Kalman filter holds a belief of what the current power profile is and predicts the temperature profile based on this belief. Meanwhile, the thermal sensors keep measuring the temperature. The power estimation method in [96] iteratively compares the temperature predicted by Kalman filter and sensor observations. If the error between them is close to zero, this indicates that the belief of current power state is correct. Otherwise, the belief might be wrong, which means the power state has changed. Once the change in power state is detected, it tries to decide the new power state, which is the one most likely to result in the current sensor reading. Interested readers are referred to [96] for the details of this adaptive power estimation approach.

Once the power profile is obtained, we select the best pressure drop which provides enough cooling for this power profile using minimum pumping power. Hence the micro-channel based DTM problem is formally stated as follows.

Given: a 3D-IC design, its power distribution is a function of the architecture and application. Assuming the power profiles are given (or estimated using appropriate sensors) and the micro-channel structure is also fixed, we would like to find

the pressure drop for each power profile such that the temperature across the chip is within acceptable limits while minimizing pumping power:

$$\begin{aligned} min \quad & P_{pump}(\Delta p) \quad \Leftrightarrow \quad min \quad \Delta p \\ s.t. \quad & \mathbf{G}(\Delta p) \cdot \vec{T} = \vec{P} \\ & \vec{T} \leq T_{max} \\ & \Delta p_{min} \leq \Delta p \leq \Delta p_{max} \end{aligned} \tag{3.18}$$

The objective minimizes the pumping power used by micro-channels. When the regular straight micro-channels are used, the pumping power can be calculated using Equation 2.16. If bended micro-channels are used, the pumping power is calculated using Equations 2.14 and 2.21.

The first constraint indicates the resistive thermal model, where $\vec{P}$ is a 3D-IC power profile and $\vec{T}$ is the corresponding thermal profile, and $\mathbf{G}$ is the thermal conductivity matrix which depends on the pressure drop Δp. The second constraint indicates that the peak temperature should not exceed the thermal constraint T_{max}. The last constraint gives the feasible range of pressure drop.

This optimization problem is difficult to solve directly because of the complexity of thermal model and the impact of micro-channel on temperature. Therefore we use a linear search based approach to find the best pressure drop. Assume the micro-channel structure is already decided, therefore the pumping power P_{pump} is only a function of pressure drop in this problem. It can be proved that the pumping power for both straight and bended micro-channels is monotonic increasing func-

tion of the pressure drop Δp. Hence minimizing pressure drop basically minimizes pumping power, and the problem is simplified to finding the minimum pressure drop that provides enough cooling.

The pressure drop Δp influences the heat resistance R_{heat}, thereby changing the cooling performance. Increase in pressure drop results in increased fluid velocity v and flow rate f, while higher flow rate results in smaller heat resistance R_{heat} and hence better cooling performance.

In summary, a larger pressure drop would result in better cooling at the cost of higher pumping power. Hence cooling effectiveness is a monotonic function of pressure drop. Therefore the linear search approach can find the best pressure drop. Specifically, this is done by starting from the minimum pressure drop $\Delta p = \Delta p_{min}$ and increasing it step by step until thermal constraint is satisfied. Due to the monotonic nature of the impact of pressure drop on micro-channel cooling effectiveness, this linear search approach can result in the optimal selection of pressure drop for a given micro-channel configuration.

3.8.2 Performance of Micro-channel Based DTM

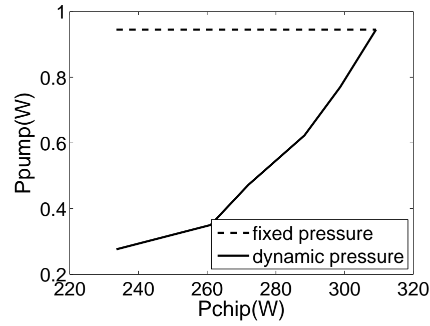
We then implemented the runtime thermal management by micro-channel pressure drop control. Here we assume the underlying micro-channel design is the non-uniform straight micro-channel configuration proposed in Section 3.3. We use the same 3D structure as Section 3.7 and tested three groups of benchmarks with different power profiles. In the first group (*group L*), we generate 6 different 3D-IC

power profiles whose total dissipated power (TDP) ranges from $220 - 320$W. Based on the non-uniform micro-channel design, we select the best pressure drop for each power profile and calculate the associated pumping power. The second (*group M*) and third (*group H*) groups are generated in a similar way, but with higher total dissipated power.

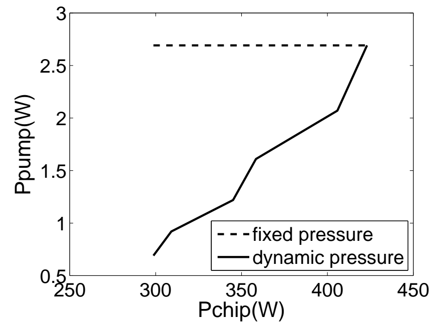
Figure 3.18 shows the required pumping power for each group of benchmarks using runtime DTM and fixed pressure drop approach. In fixed pressure drop approach, we use the lowest pressure drop that could provide enough cooling to all benchmarks in this group. P_{chip} is the TDP of each benchmark. The runtime pressure drop controlling approach achieves an average of 39%, 43% and 46% pumping power saving for benchmark groups L, M and H. The pressure drop calculation can be done off line and stored in a table. Once we detect a specific power profile occurs, we simply look up the best pressure drop for this power profile.

3.9 Summary

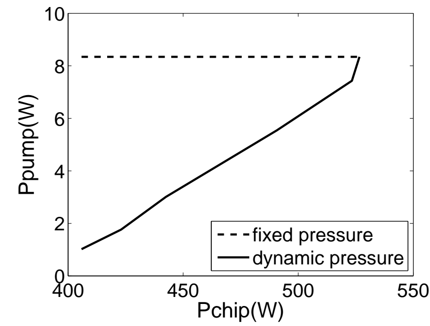
This chapter investigated the optimized micro-fluidic cooling configurations. The first configuration (hotspot-optimized non-uniform micro-channel design) allocates micro-channels only in hotspot regions so that less channels are used, thereby saving pumping power. In this configuration, straight micro-channels are used. The straight micro-channels are easy to manufacture and more power efficient compared with bended micro-channels of the same length. However, straight micro-channels are inefficient in addressing the spatial constraints imposed by TSVs. Hence in the



(a)



(b)



(c)

Figure 3.18: Runtime pressure drop control versus fixed pressure drop for (a) group L, (b) group M, (c) group H

second configuration, we proposed the usage of bended micro-channel, which can be flexibly routed to hotspot regions while avoiding TSVs. In order to further reduce the pumping power overhead, we also proposed a hybrid cooling network which utilizes dummy thermal TSVs (that reinforce vertical heat transfer) and micro-channels together. Compared with the conventional micro-channel design that spreads straight micro-channels all over the interlayer region, the optimized configurations can result in 57%, 68% and 78% pumping power savings respectively. In these designs, micro-channel structures are designed after the electrical part of the chip, hence they are compatible with the standard IC design flow.

We also proposed a micro-channel based dynamic thermal management method that controls the pressure drop at runtime to allow real time thermal control. Through runtime pressure drop tuning, we can further save about 43% pumping power compared with using fixed pressure drop.

However, as illustrated in Section 1.4, the electrical, thermal, reliability and cooling aspects are all interdependent. Hence, separating the design of electrical and cooling system will lead to sub-optimal designs. In the next chapter, we will investigate the electrical and cooling system co-design to achieve further power-performance improvement.

Chapter 4

Co-design of Electrical and Fluidic Cooling Systems

4.1 Motivation for Co-Design

In the conventional chip design flow, cooling considerations are put in place after the entire system has been designed (as Figure 4.1 shows). Such a *postfix* approach can lead to sub-optimality, such as significant pumping power, competition with TSVs, thickening of silicon substrate and impact on reliability.

As illustrated in Section 1.4, the electrical, thermal, reliability and cooling aspects are all interdependent. It is important to investigate the interplay between electrical and fluidic aspects, and develop avenues for co-design. Such co-design can result in the following advantages:

1. Higher cooling in timing critical areas results in better performing designs since transistor delay is proportional to temperature.

2. Higher cooling in timing critical areas enables us to aggressively pursue high power dissipating performance enhancements such as increasing supply voltage. This results in higher performance without impacting temperature since the extra heat can be manager by micro-fluidics.

3. The design optimization could be more aggressive since temperature issue can be addressed by aggressive cooling (placement, floorplanning etc.)

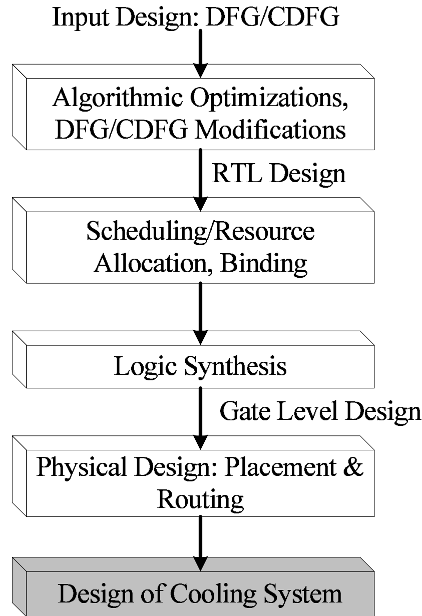


Figure 4.1: Conventional chip design flow

4. Increasing the cooling levels in high leakage areas helps reduce the overall power since leakage is a highly non-linear function of temperature. Reduction in leakage may be significant enough to make increase in pumping power irrelevant.

5. Micro-fluidics may impact silicon thickness causing TSV performance degradation. By smart electrical design, this degradation could potentially be removed. For example, degradation in TSV performance could be overcome by stronger drivers.

In this chapter, we investigate two electrical and cooling co-design problems. Section 4.2 investigates the TSV allocation/assignment and micro-channel placement co-design [70], and in Section 4.3, a gate sizing and micro-fluidic co-design problem is investigated [71].

4.2 Co-optimization of TSV Assignment and Micro-Channel Placement

In 3D-ICs, the interlayer nets use TSVs to deliver signals and power among different layers. Recently, significant attention has been made to the problem of allocating interlayer nets to TSVs that allow their successful routing. Existing work mostly tries to address this problem with the objective of minimizing total wirelength. Two general approaches have been investigated: Post-Placement [48][90][95] and In-Placement [36]. In Post-Placement approaches, cells are firstly placed in the 3D-IC. This determines the whitespace distribution capable of supporting TSVs. These potential TSV locations are then allocated to the interlayer nets such that the total wirelength is minimized [48][90][95]. In-Placement approaches perform simultaneous optimization of cell placement, TSV placement and interlayer net to TSV assignment during the 3D-IC placement process itself. While both approaches have their advantages, in our work, we assume the placement to be already done before TSV assignment to the interlayer nets (Post-Placement paradigm), though our work could also be extended to the In-Placement approach.

Conventional Post-Placement approaches for interlayer net to TSV assignment do not consider the possibility of adding micro-channels in the interlayer regions. TSVs impose significant constraints on how and where the micro-channels can be located, and form obstacles to the micro-channel placement since the micro-channels cannot be placed at the locations of TSVs. The location of TSVs is essentially decided by the allocation of interlayer nets to TSVs. The exiting works for Post-

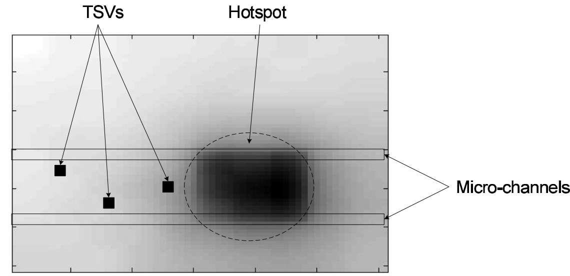


Figure 4.2: Thermal profile of one 3D-IC layer, and an example of TSV and micro-channel allocation where TSVs constraint us from allocating micro-channels at hotspots

Placement TSV allocation (which ignore the possibility of allocating channels) and micro-channel placement as proposed in the previous chapter (which assume the TSV locations to be fixed) do not consider the possibility of combining these steps for obtaining better results.

Two trivial approaches for allocating TSVs to nets and micro-channels to interlayer regions *together* can be conceived as follows: TSV first approach and Micro-channel first approach. If micro-channels are allocated before TSVs, there is a possibility of increase in wirelength since the available whitespace for TSVs shrinks due to the existence of micro-channels which deter allocation of TSVs in those areas. A TSV first approach also has disadvantages. For instance, if TSVs are placed at or near hotspot regions which preventing the allocation of micro-channels at that hotspot, the cooling effectiveness of micro-channels will suffer.

In this section, we investigate co-optimization of TSV assignment and micro-channel allocation simultaneously such that the total wirelength is minimized, and maximizing the micro-channel cooling effectiveness [70]. As stated earlier, we assume a Post-Placement paradigm.

4.2.1 Problem Formulation

The problem is stated in Table 4.1. The objective minimizes a combination of the cooling power required by micro-channels and the total wirelength used by all interlayer nets. It is noteworthy that an interlayer net is allocated to a *set* of TSVs since several TSVs spanning multiple layers may be needed to connect the source-destination pairs. The co-optimization of micro-channel allocation and TSV assignment is complex due to its discrete nature and the complexity of thermal estimation. Hence, we focus on developing effective heuristics that exploit specific mathematical properties present in this problem.

4.2.2 Algorithm for TSV Assignment and Micro-channel Placement Co-optimization

4.2.2.1 Overall Design Flow

The overall design flow is shown in Figure 4.3. We use multi-commodity min-cost flow to formulate/solve some critical aspects of the problem, hence we call this approach MCMCF. In Section 4.2.3 we discuss simplifications to this formulation that enable us to solve the problem efficiently. We firstly find the thermal criticality of all grid locations in the 3D-IC chip using a full chip thermal analysis assuming there are no micro-channels. Also, based on the 3D-IC structure and placement, we identify all the potential locations of micro-channels and TSVs.

Assuming the 3D-IC is divided into small grids (i, j, k), with i, j representing

Table 4.1: Problem formulation

Given:	
I.1:	A 3D-IC placed netlist. The placement information can be used to generate potential TSV locations;
I.2:	A netlist that describes a set of interlayer nets;
I.3:	The power profile of 3D-IC;
I.4:	A set of potential locations for interlayer micro-channels. These channels are to be incorporated in the interlayer region of the chip;
We would like to:	
O.1:	Decide the locations of TSVs;
O.2:	Assign a set of TSVs to each interlayer net;
O.3:	Decide the number and locations of micro-channels;
In such a way that:	
C.1:	The assigned set of TSVs for each interlayer net forms a path connecting the source and destination terminals of the net;
C.2:	The locations of micro-channel and TSVs do not conflict (see Figure 2.1 for detail);
C.3:	The micro-channels provide sufficient cooling for the 3D-IC, i.e. $T_i \leq T_{max}, \forall locations : i$;
C.4:	The total wirelength and required pumping power by micro-channels is minimized: $\min u_1 N + u_2 \sum_{\forall r} WL_r$ where N is the number of channels and WL_r is the bounding box wirelength of the r-th interlayer net which depends on the TSV set it has been allocated to. Constants u_1 and u_2 could be allocated based on preference for a particular tradeoff.

the face of the 3D-IC and k representing the longitudinal direction along which the micro-channel runs. The location of a micro-channel is basically the (i, j)-th grid where the channel is located. In the k-th direction, the channel spans the chip anyway. The TSV could be identified by the (i, j, k)-th grid it is located at. After initial thermal analysis, we define a thermal criticality of each potential micro-channel which basically represents the demand of allocating a micro-channel at that location. The criticality factor $c(i, j)$ for each micro-channel location (i, j) is defined as:

$$c(i,j) = \sum_{k=0}^{K} w(i,j,k) \cdot max[0, T_{i,j,k} - T_{max}] \tag{4.1}$$

where $T_{i,j,k}$ represents the temperature at grid (i,j,k), T_{max} is the maximum thermal constraint. Parameter $w(i,j,k)$ represents the thermal significance of a certain grid, and K is the number of grids in the longitudinal direction in which the channel spans the entire chip. Based on the criticality factor $c(i,j)$, we formulate the MCMCF problem and obtain the TSV assignment and micro-channel allocation simultaneously (see Figure 4.3). Thermal analysis and 2D routing are then conducted to evaluate the performance of the resulting design. If the design results in thermal violation, ends up having significant wirelength or placing too many micro-channels in some locations (this will increase cooling power and might also degrade wirelength), we will refine the criticality factor $c(i,j)$ (increase or decrease $c(i,j)$ accordingly) and re-solve the MCMCF problem.

We repeat this process iteratively until obtaining a design that achieves required tradeoff between cooling power and wirelength.

4.2.2.2 Multi-commodity Minimum Cost Flow Formulation

Given: a) the 3D-IC structure, b) potential locations of TSVs and micro-channels, c) the interlayer netlist and d) criticality factor $c(i,j)$, the multi-commodity min-cost flow (MCMCF) problem is illustrated in Figures 4.4 and 4.5. Figure 4.4 illustrates a 3D-IC with three active layers and two interlayer nets along with four potential TSVs. The potential locations of micro-channels have also been indicated.

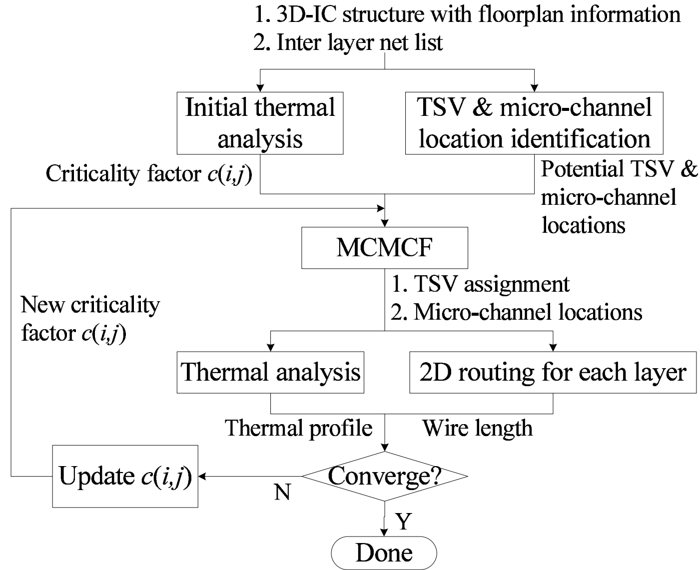


Figure 4.3: Overall design flow of MCMCF based algorithm

Both front and top views have been illustrated that indicate TSV and net locations in the 3D-IC grids. Our objective is to find the allocation of nets to TSVs and micro-channels such that cumulative objective indicated in the previous section is minimized: $\min u_1 N + u_2 \sum_{\forall r} WL_r$. Assuming u_1 and u_2 are the same for the sake of ease in exposition, we instantiate a multi-commodity min-cost flow formulation as follows.

For each net, we allocate one unit of unique commodity flow. Hence J nets would correspond to J distinct units of commodity flows. The flow network has one node for each terminal of the nets and also the potential TSV locations. We assume that the net terminal in the higher layer is the source of this one unit flow and net terminal in the lower layer is the sink. We assume all nets are two terminal. For the example shown in Figure 4.4, the flow network is illustrated in Figure 4.5 which indicates that $net1$ and $net2$ terminals in the top layer are sources. They are connected by directional edges to the TSV nodes in that active layer. If the nets

span multiple (more than 2) layers, then the TSVs in this layer would connect to the TSVs in the layer just below to transfer the signal. This is also indicated in Figure 4.5(a) where TSVs in layer 1 are connected by directional edges to TSVs in layer 2. Finally destination or sink terminals of nets are also connected by directional edges to TSVs in that layer as indicated in the figure. Note that the edges always carry flow from source to sinks. Also, by construction this network forms a directed acyclic graph. Let us ignore the presence of micro-channels for now (for ease of explanation). The problem of allocating nets to TSVs could be modeled as a multi-commodity min-cost flow (and the flow graph is illustrated in Figure 4.5(a)). Let each net to TSV edge or TSV to TSV edge have a cost which is simply the half perimeter bounding box between the two. Let each TSV node have a total capacity of 1. Also all edges have an individual commodity capacity as 1 and a total capacity as 1. The multi-commodity min-cost flow solution that sends the unique commodity flow from each net source to the corresponding net sink on the network in Figure 4.5(a) at the minimum total cost corresponds to the net to TSV assignment with minimum total wirelength. A total unity capacity for each TSV node ensures that only one net is allocated to it.

Now we extend this formulation to account for the presence of micro-channels. As indicated in Figure 4.4, some micro-channel locations conflict with TSVs while others don't. The micro-channel is allocated an additional flow commodity. Hence if there are J interlayer nets then the total commodities becomes $J + 1$. Figure 4.5(b) indicates the process of accounting for micro-channels in the flow network of Figure 4.5(a). The figure shows the top view of an interlayer region where the micro-

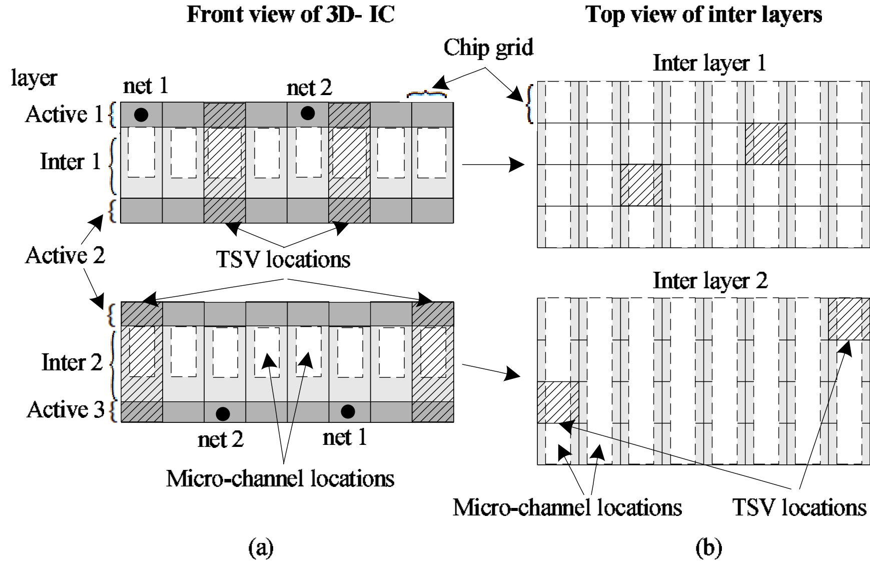


Figure 4.4: 3D-IC with potential TSV and micro-channel locations

channels are located and potentially conflict with TSVs. The micro-channels span the entire length of the chip in the k direction. If there is even one TSV allocated in this path, a channel cannot be allocated and vice versa. Now some potential micro-channel locations do not have any potential TSVs while others do (see Figure 4.5(b)). We instantiate a source at the beginning of each potential channel location and a sink at the end. This source contains a unique commodity corresponding to the fluid flow. Note that all sources corresponding to micro-channel locations have the same flow type (same commodity). Several paths exist between the source and sink for a particular channel location. The simplest path is the one that goes through all the grids that span the entire length of the chip (see Figure 4.5(b)). Some of these grids are potential TSV locations and have other nets and/or TSVs connected to them by way of directional edges (as indicated in Figure 4.5(a)). The fluid flow edges are directed longitudinally while the net interconnection edges are directed vertically. As indicated earlier, the TSV nodes have a total capacity of

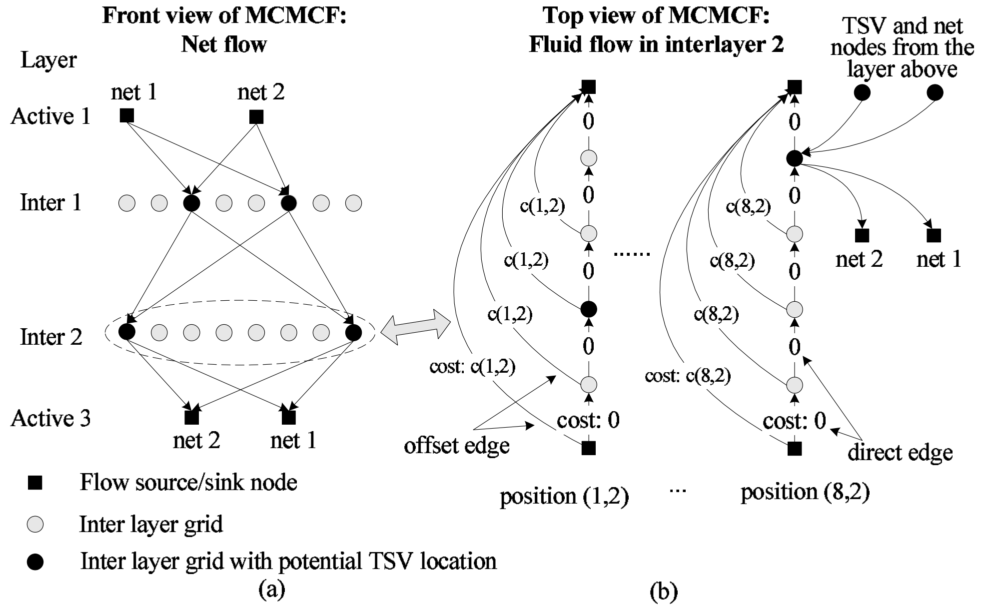


Figure 4.5: Multi-commodity min-cost flow formulation

one. The longitudinal edges that represent fluid flow edges have unit capacity for the fluid flow commodity while 0 capacity for all the J net commodities. The fluid flow edges that connect the adjacent grids (*direct edges* in Figure 4.5(b)) have a cost of 0. Now each intermediate grid in this direct path is also connected directly to the fluid sink for that channel location by way of *offset edges* as well. This edge also has a fluid commodity capacity of 1 and net commodity capacity of 0. The cost of this edge is $c(i, j)$ which represents the cooling demand for that channel location. All the edges that represent net to TSV or TSV to TSV connection have a fluid flow commodity capacity as 0. Note that different micro-channel locations do not interfere since the network does not have any edges that interconnect them. Now the cheapest way of sending a fluid flow commodity from source to sink is to follow the simple path that spans all the adjacent nodes. The cost of this path is 0 but it forces us to use all the potential TSV nodes on the path since they have a

total capacity as 1. Hence none of these potential TSV nodes can be used for net interconnection. On the other hand, if any one of these nodes has been allocated to a net, a fluid commodity cannot go through this simple path anymore and it has to take any one of the alternative paths to the sink (see *offset edges* in Figure 4.5(b)). Any such alternative path has a cost of $c(i, j)$ which represents the price that we pay by *not* having a channel at that location since we would rather use some of the TSVs on the channel path for routing nets.

Sending min-cost multi-commodity flow on this network results in an allocation of nets to TSVs and micro-channels to channel locations such that the total cost is minimized. This cost is a combination of $c(i, j)$ and bounding box wirelength, and represents a balance between cooling and wirelength. Solving multi-commodity problem is a challenging problem since the formulation is generally NP-Complete, although several effective heuristics have been developed. In Section 4.2.3, we investigate some specific properties in our problem that help us simplify the formulation thereby enabling us to use simpler, computationally efficient heuristics.

4.2.2.3 Iterative Optimization

As indicated in Figure 4.3, once an allocation of micro-channels and TSVs has been conducted, we perform: a) routing to compute the actual wirelengths, and b) thermal analysis. If the wire-lengths are unacceptable, thermal violations occur or the system is overcooled (pumping power is wasted), the $c(i, j)$ values are re-allocated and the problem re-solved. If wire-lengths are very high, then $c(i, j)$

values are uniformly scaled down enabling us to prefer wirelength over channels. If the system experiences thermal violations, $c(i,j)$ values are increased enabling us to use more micro-channels. If the system is non-uniformly overcooled then regions where excessive cooling is available are subjected to a reduction in $c(i,j)$ which could end up in removing the channels in favor of using TSVs. Such an approach assists in achieving the optimal balance between wirelength and cooling power while satisfying the thermal and interconnection constraints.

4.2.3 Computational Simplifications

4.2.3.1 Multi Layer Case

Solving multi-commodity flow instances in general is computationally intractable. For our specific case, this is a bigger issue since the number of commodities is linear in the number of interlayer nets J (which could be quite large). This significantly adds to the number of unknowns in the problem formulation making its solving computationally expensive. We first simplify the formulation without losing optimality followed by effective heuristics. We transform the flow graph illustrated in Figure 4.5 to the one illustrated in Figure 4.6. For the moment let us ignore the fluid flow network in Figure 4.5(b). For each distinct net, let us replicate the entire network graph in Figure 4.5(a) J times (one replica for each net). This is illustrated in Figure 4.6(a). Basically all the TSV nodes in the original network appears J times in the new network. The graphs for each net do not have any common edges, hence we don't need to represent the net flows by different commodities. All the

net flows belong to the same commodity. The edge costs and the node/edge capacities are exactly the same as before. Sending unit commodity min-cost flow on this network, though, does not solve our problem. This is because the same TSV may be used by two or more nets. In order to address this problem we can allocate a *bundle capacity* to all replicated TSV nodes corresponding to the same TSV. A bundle capacity constraint in network flow problems allocates a total capacity to a bundle of nodes or edges. In our case we can set a bundle capacity constraint of 1 to all replicated TSV nodes belonging to the same TSV. This is illustrated in Figure 4.6(a). The problem continues to be NP complete but we have eliminated the need for different commodities by adding an additional bundle constraint. We found through our experiments that this significantly enhanced the computational efficiency.

Adding micro-channel allocation constraints is illustrated in Figure 4.6(b). Just as Figure 4.5(b), each potential micro-channel location has the associated network as illustrated, but the TSV nodes in the fluid network do not have the edge connections to net flow in this case. Instead of allocating a different commodity to fluid flow, we allocate the same commodity. Now the TSV location nodes in the network of Figure 4.6(b) have a bundle capacity of 1 with the replicated TSV nodes for the corresponding TSV in the rest of the network. Hence if a TSV is allocated to a net, then it cannot be allocated to any other net or micro-channel. The problem now becomes a (single-commodity) min-cost flow problem with bundle capacity constraints. While solving this problem formulation is NP Complete, it has significantly smaller number of unknowns although the constraints are a bit

more complex. We solve this problem by assuming that the discrete flow variables are continuous. This results in a linear programming approximation (polynomially solvable) for this discrete problem. After getting the solution, non-discrete values are rounded up appropriately to give a valid solution.

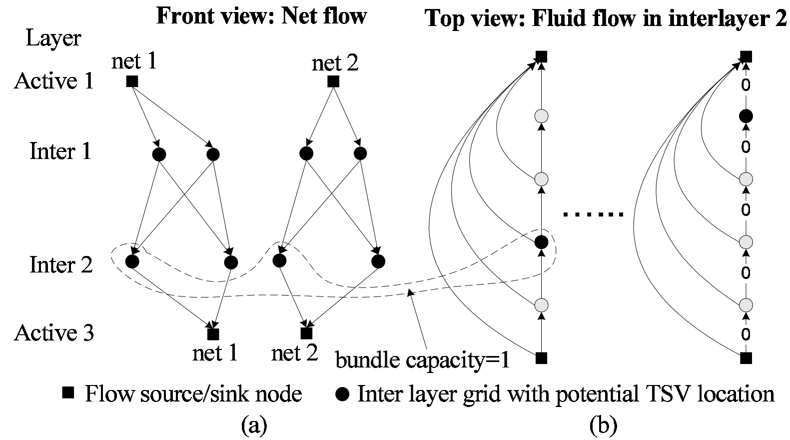


Figure 4.6: Computationally simplifying transformation for multi-layer case

4.2.3.2 Two Layer Case

Now we discuss the special case where there are only two active layers stacked together. While the simplification for multi-layer case described above could certainly be applied here, there are additional transformations we can use. Consider the instance illustrated in Figure 4.7(a) where we have two nets and two TSVs. Once again, let us ignore the micro-channel constraints for the moment. Allocation of nets to TSVs in this case is easier than the multi-layer case since it can be transformed to a simple case of bipartite matching. We instantiate a network as illustrated in Figure 4.7(b). For each net (unlike net terminal in the previous case) we have a node and for each TSV we have a node. We have directed edges between nets and TSVs whose cost is the total bounding box between the net's two terminals and the

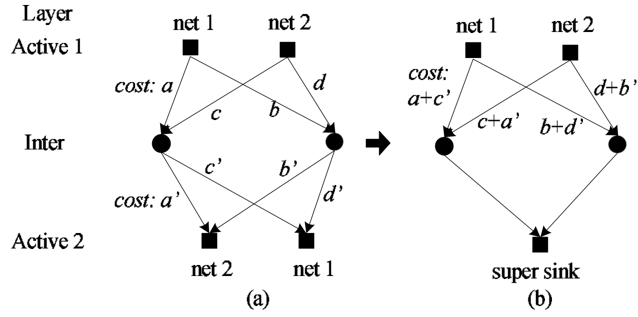


Figure 4.7: Computationally simplifying transformation for two-layer case

TSV pads in the corresponding layers (see Figure 4.7(b) for an illustration). Each node corresponding to the nets has a unit flow (of the same commodity) available. We also have a super sink that is connected to all the TSVs. The TSV nodes have a capacity of 1. Sending min-cost flow from net nodes to the super sink would essentially correspond to allocation of nets to TSVs with minimum total wirelength optimally in polynomial time. In order to add micro-channel location constraints to this formulation, we essentially apply the method used in the multi-layer case (with bundle constraints). Note that in this case, no replication of nodes for TSV assignment was needed as in the multi-layer case, hence the generated formulation is much simpler than simply applying the previous technique to this case directly. The problem is still NP Complete due to the bundle capacity constraints. We simplify the formulation to a linear program LP by assuming the flow variables are continuous. The generated continuous solution is then discretized by rounding of the non-discrete variables.

Note that for multi-pin interlayer nets, we firstly partition them into multiple two-pin nets, and use the aforementioned method to assign all the two-pin nets to TSVs.

4.2.4 Performance of TSV Assignment and Micro-channel Placement Co-design

4.2.4.1 Comparison of Wirelength and Pumping Power

In our experiment, we tested both two-layer and three-layer 3D-ICs. We use IBM-PLACE 2.0 circuits with placement information as the benchmark [2]. For each test, we choose two or three circuits from $ibm01 - ibm10$ circuits, each circuit corresponds to one 3D-IC layer. Based on the placement information, we find the whitespace between layout, which are basically the potential TSV locations. The number of potential TSV locations ranges from around 50-1000. We also randomly generate 30-200 interlayer nets. To obtain the power profiles for each layer, we randomly assign a value for each cell as the power density for the cell. The chip dimension is $9 \times 9mm^2$. The micro-channel width $\times$ height is $100 \times 200\mu m^2$, and the diameter of TSV is $10\mu m$. The maximum temperature constraint T_{max} is 85℃.

We compare the wirelength and pumping power achieved by our co-optimization approach and *TSV first*, *Micro-channel first* approaches.

1. *TSV first* approach firstly assigns TSVs to interlayer nets assuming there are no micro-channels. Once TSVs are assigned and hence TSV locations are decided, we allocate micro-channels in the remaining interlayer regions using the approach in Section 3.3;

2. *Micro-channel first* approach allocates micro-channels first assuming there are no TSVs, and then assigns interlayer nets to the remaining available TSV

locations.

For each approach, once we obtained the TSV assignment result, we route the interlayer net terminals to the TSVs (or TSVs to TSVs) in each layer separately using Labyrinth 2D router [3] to obtain the total wirelength (WL). We also estimate the pumping power P_{pump} based on the number of channels used and the given pressure drop. Table 4.2 shows the benchmark information.

Table 4.2: Benchmark Information

Ckt	# Layer	# TSV	# Interlayer nets
1	2	56	30
2	2	119	50
3	2	190	80
4	2	348	100
5	2	652	125
6	3	175	50
7	3	511	80
8	3	714	100
9	3	1111	200

Table 4.3: Comparison between our approach, TSV first and channel first approach (P_{pump} : W, $WL : m$, temperature: ^oC)

	Air	TSV first			Micro-channel first			Co-optimization						
	cool			Below			Below			Below	WL change wrt		P_{pump} change wrt	
Ckt	T_{peak}	WL	P_{pump}	T_{max}	WL	P_{pump}	T_{max}	WL	P_{pump}	T_{max}	TSV first	MC first	TSV first	MC first
1	106.46	0.19	2.13	Y	0.22	1.48	Y	0.19	1.70	Y	+0.00%	-13.64%	-20%	+14%
2	101.11	0.28	n/a	N	0.33	0.84	Y	0.28	1.03	Y	+0.00%	-15.15%	n/a	+22%
3	121.97	0.45	n/a	N	0.52	1.04	Y	0.47	1.32	Y	+4.44%	-9.62%	n/a	+27%
4	110.25	1.25	6.81	Y	1.33	1.70	Y	1.26	2.55	Y	+0.80%	-5.26%	-63%	+50%
5	128.83	1.53	5.11	Y	1.63	1.70	Y	1.52	2.55	Y	-0.65%	-6.75%	-50%	+50%
6	135.28	0.37	n/a	N	0.43	4.25	Y	0.37	5.10	Y	+0.00%	-13.95%	n/a	+20%
7	154.06	0.91	20.42	Y	0.98	5.53	Y	0.92	5.95	Y	+1.10%	-6.12%	-71%	+8%
8	152.39	1.55	9.36	Y	1.63	6.04	Y	1.51	6.80	Y	-2.58%	-7.36%	-27%	+12%
9	161.05	2.99	10.21	Y	3.22	5.29	Y	3.00	5.95	Y	+0.33%	-6.83%	-42%	+12%
Avg											+0.38%	-9.41%	-46%	+24%

Table 4.3 shows the comparison of wirelength and micro-channel cooling power

for the three approaches. In the table, "below T_{max}" indicates if the achieved thermal profile satisfies the thermal constraint, *MC first* indicates the *micro-channel first* approach. Table 4.3 shows that using air cooling results in thermal violation for all power profiles, while micro-channels can provide sufficient cooling. Moreover, using *TSV first* approach, though achieves good wirelength compared with *micro-channel first* approach, uses about 160% more pumping power since the existence of TSVs deters the optimal allocation of micro-channels. Furthermore, for some benchmarks, the TSVs are allocated in thermal critical regions, in which cases micro-channels cannot effectively cool these thermal critical regions thereby causing thermal violations. On the contrary, *micro-channel first* approach, though saves pumping power, results in up to 15% wirelength increase compared with *TSV first* approach. Our approach considers both wirelength and pumping power simultaneously. The wirelength increase in our approach compared with *TSV first* approach is only 0.38%, while compared with *MC first* approach, our approach saves 9.41% wirelength. In some benchmarks, our approach even results in slightly better WL than *TSV first* approach, this is because both approaches use the bounding box wirelength (which basically gives a lower bound of the routing wirelength) when solving the TSV assignment problem, while the real routing result also depends on the relative positions between interlayer net terminals and TSV locations. Therefore, in these benchmarks, although our approach results in slight degradation in bounding box wirelength, its real routing wirelength is better than *TSV first* approach. Comparing the micro-channel pumping power, our approach achieves 46% pumping power savings compared with *TSV first* approach, and uses 24% more

pumping power compared with *micro-channel first* approach. Moreover, for benchmarks where thermal violations occur using *TSV first* approach, using our approach could reduce the temperature below thermal constraints without consuming excessive pumping power.

4.2.4.2 Tradeoff Between Wirelength and Pumping Power

The value of criticality factor $c(i, j)$ could be adjusted to control the weight between wirelength and cooling provided by micro-channels. Usually, decrease in pumping power is at the cost of increased wirelength, and vice versa. Such tradeoff is illustrated in Figure 4.8, which shows the wirelength versus pumping power for one benchmark (all data points satisfy the thermal constraints). When thermal violations occur, more efficient allocation of micro-channels could be adopted by sacrificing some wirelength, or more channels are allocated in the unused regions surrounding the hotspot which leads to an increase in pumping power. When pumping power is too high, we could try to better allocate micro-channels to improve its cooling effectiveness at a cost of longer wirelength. When wirelength is more preferable, we could assign TSVs towards further reduction of wirelength while sacrificing micro-channel cooling effectiveness (leading to higher pumping power).

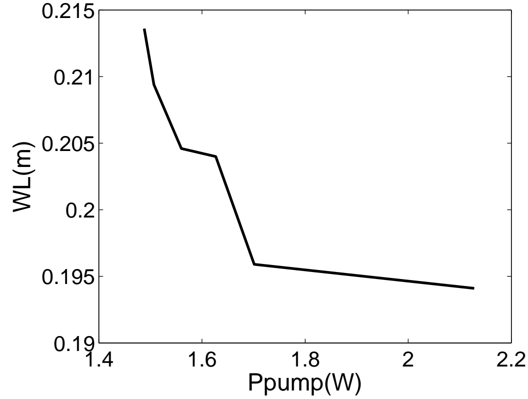


Figure 4.8: Tradeoff between wirelength and pumping power

4.3 Co-optimization of Gate Sizing and Micro-Fluidic Cooling

4.3.1 Motivation of Simultaneous Gate Sizing and Micro-channel Distribution

Distribution of channels in the interlayer region (deciding the channel placement) can be controlled to favor some sub-regions over others. As investigated in the previous chapter, the distribution of channels can be used to control the local temperature of 3D-IC subregions, unlike conventional air cooling where no such control is possible. This *localized thermal control* enabled by apt distribution of channels (higher channel counts in some areas over lower channel counts in others) offers several advantages to the 3D-IC design process, which are ignored by the conventional *postfix* approach for design of the cooling system.

The power, performance and temperature aspects of 3D-ICs have a very complex interdependence. Temperature profile depends on both the amount as well as distribution of power. Non-linear leakage thermal interdependence implies higher temperatures leading to greater power. Higher temperature also impacts the device

performance. Addressing these complex interdependencies between power, temperature and performance has been a major focus of research both for 2D and 3D ICs. Localized temperature control enabled by micro-channel distribution can be exploited in a number of ways by the 3D-IC design optimization process.

1. **Improving the circuit speed:** Allocation of greater cooling surrounding timing critical areas could be used by 3D-IC design methods to improve timing further by aggressive timing optimization since the associated power dissipation could be addressed by greater cooling. Reduced temperatures would also contribute to an overall speeding up of circuit.

2. **Reducing dynamic and leakage power dissipation:** Greater cooling in high leakage areas would directly reduce their leakage levels due to non-linear dependence between leakage and temperature. Reduction in temperature around timing critical circuits would result in an overall speeding up of the design. Hence we do not need aggressive timing optimization helping save both dynamic and leakage power. Reduction in power would further reduce temperatures causing a favorable positive feedback. The reduction in power dissipation may be significantly greater than an increase in the pumping power (experimental results to support this claim would be provided subsequently). Hence the total power of the 3D-IC including dynamic, leakage and pumping would be reduced.

3. **Reduction in pumping power:** Design of 3D-IC would decide the location and nature of hotspots and nature of power dissipation. Co-optimization of

the 3D-IC system and the channel distribution could be used to simplify the cooling configuration and therefore save pumping power.

4. **Fundamental advancement in power-performance tradeoff:** Per the advantages noted above, co-optimization of cooling and the 3D-IC design enables better performance under a given power envelope and better power for a given performance constraint, thereby resulting in fundamental improvement in power-performance tradeoff. Experimental data to support this claim is illustrated subsequently.

Overall, it can be seen that there is sufficient motivation for co-optimization of the 3D-IC physical design as well as distribution of channels. Co-design of 3D-IC and the fluidic cooling infrastructure can fundamentally improve the power performance tradeoff in 3D-ICs. In this section we attempt to highlight the need for this co-design and the associated challenges and opportunities. We investigate the simultaneous gate sizing and micro-channel distribution problem in 3D-ICs as an illustration of the advantages of this co-optimization [71].

4.3.2 Modeling of Gate Delay

The maximum delay of circuit is usually decided by the latency of critical pathes, which is largely influenced by the delay of gates on these critical pathes. The gate delay is influenced by many parameters, such as the gate size, carrier mobility, and threshold voltage, etc.

Many works model the gate delay as a posynomial function of the gate sizes

as:

$$d_i \propto \eta_{0i} + \frac{\sum\limits_{\forall k \in FO(g_i)} \eta_{ki} \cdot s_k}{s_i} \tag{4.2}$$

Here s_i is the width of gate g_i, and $s_{k,\forall k \in FO(g_i)}$ are the sizes of all gate g_i's fanouts [35]. This model shows that the gate delay is a monotonically decreasing function of its own size, but a monotonically increasing function of the sizes of its fanout gates. Therefore, increase in the size of gate g_i can result in a reduction in g_i's delay, however this would increase the delay of gate g_i's fanin gates.

Some of the circuit parameters, such as the threshold voltage and mobility are sensitive to temperature [86]. [47] models the dependency of gate delay on temperature as a polynomial function:

$$d_i \propto T_i^{\sigma}, \qquad \sigma \approx 1.19 \tag{4.3}$$

By incorporating impact of both gate sizes and temperature, we can model the gate delay as a function of gate sizes and temperature:

$$d_i \propto T_i^{\sigma} \cdot (\eta_{0i} + \frac{\sum\limits_{\forall k \in FO(i)} \eta_{ki} \cdot s_k}{s_i}) \tag{4.4}$$

Here s_i, T_i are the width and temperature of gate g_i, s_k is the width of g_i's fanout gates, σ, η_{0i} and η_{ki} are constants.

This model shows that change in the following parameters can result in gate delay reduction: (a) increase of its own width, (b) decrease in the width of its fanouts, and (c) reduction in gate temperature.

4.3.3 Problem Formulation

The problem of gate sizing and micro-channel placement co-optimization is formally stated as follows. Given a 3D-IC circuit and the associated gate and TSV placement (as Figure 3.1 shows), we would like to decide the size of all gates and location of interlayer micro-channels such that the total power consumption (including the dynamic and leakage power, as well as the pumping power consumed by micro-channels) is minimized, while at the same time minimizing the longest path delay and ensuring silicon temperature to be less than the maximum constraint. The channels should not come in conflict with TSVs, which have been placed already. The co-optimization problem is formulated in Equation 4.5. Here we assume that gates and TSVs have been placed on a grid (each gate/TSV is within a grid). Also the gate sizing does not change the gate's grid location. Note that these assumptions are similar to other works dealing with in-place gate sizing.

$$
\begin{aligned}
&\textbf{Decision variables}: \vec{s},\ \mathbf{B} \\
&min \quad \sum_{\forall gate: g_i} (P_{d,i} + P_{l,i}) + P_{pump} \\
&s.t. \quad 1. \quad t_j + d_i(\vec{s}, T_i) \leq t_i, \forall \text{gate } g_i, g_j \in FI(g_i) \\
&\qquad\ \ 2. \quad t_i < t_{con}, \forall \text{gate } g_i \in PO \\
&\qquad\ \ 3. \quad \mathbf{G}(\mathbf{B}) \cdot \vec{T} = \vec{P}(\vec{s}, F, \vec{T}) \\
&\qquad\ \ 4. \quad 0 \leq \vec{T} \leq \vec{T}_{max} \\
&\qquad\ \ 5. \quad s_{min} \leq s_i \leq s_{max}, \forall \text{gate } g_i
\end{aligned}
\tag{4.5}
$$

The decision variables in this problem are the gates size $\vec{s}$ and micro-channel locations $\mathbf{B}$.

The objective of the optimization problem is to minimize the total power consumption of the 3D-IC (including dynamic, leakage and pumping power) for the given timing constraint t_{con}. Here $P_{d,i}$ and $P_{l,i}$ represent the dynamic and leakage power of gate g_i, which can be calculated based on the models in Sections 2.4.1 and 2.4.2. The dynamic power depends on the gate sizes $\vec{s}$ and clock frequency F, and leakage power depends on both gate sizes $\vec{s}$ and thermal profile $\vec{T}$ (temperature in all grids). The clock frequency is usually decided by the maximum circuit delay. Hence, in this work, we assume the clock frequency is the inverse of timing constraint $F = 1/t_{con}$.

The first two constraints are timing constraints, indicating that the signal propagation delay from the primary inputs (PIs) to primary outputs (POs) should be within the timing constraint t_{con}. Here t_i denotes the signal arrival time at the output of gate g_i from the primary inputs and d_i is the propagation delay of gate g_i. The delay, which depends on gate sizes and temperature, is calculated using the model in Equation 4.3.2. We assume the 3D-IC is divided into grids. For ease of explanation, we assume each grid only contains one gate. Hence grid i contains gate g_i and has the temperature T_i. If a grid does not have a gate, the corresponding power is 0 and the temperature would be decided by neighboring grids based on the conductivity matrix $\mathbf{G}$. The 3D-IC thermal profile $\vec{T}$ is then represented by the temperature of all grids: $\vec{T} = \{T_{i,\forall grids:i}\}$. Note that this formulation is easily extendable to the case where each grid contains multiple gates.

The third constraint indicates the interdependency between temperature and power. Let $\vec{T}$ and $\vec{P}(\vec{s}, F, \vec{T})$ represent the thermal and power profiles at all grids i in 3D-IC. The power dissipated in a grid i is $P_i = P_{d,i} + P_{l,i}$ (if a grid does not have any gate then its power is 0). Note that the power profile is a function of gate sizes and temperatures. Here **G** represents the 3D-IC conductivity matrix which depends on the properties of the material, TSVs as well as design of the micro-channel structure **B**. The last two constraints are the maximum temperature constraint and feasible gate size range.

The power, temperature and gate delay are interdependent in a complex way, making this co-optimization problem difficult to solve. The allocation of micro-channels at discrete locations adds further complexity to this problem.

4.3.4 Algorithm for Gate Sizing and Micro-channel Placement Co-optimization

The problem formulation illustrated above is quite complex. We develop an iterative optimization approach where each step systematically solves some aspects of the problem. We have strived to use rigorous optimization methods as much as possible. Fundamentally the overall optimization problem is decomposed into two: deciding the gate sizes and grid temperatures simultaneously and then designing the micro-channel distribution which removes the heat generated by the circuit (function of temperature and gate size) while coming as close as possible to the prescribed temperature. This process is iterated several times as summarized below.

Step 1: Ideal heat sink and gate size co-optimization: We first simplify the problem by assuming that temperature in each grid is perfectly controllable and is not dependent on the 3D-IC conductivity matrix **G**. The resulting solution allocates a gate size and temperature level to each gate/grid. The ideal case acts as a guideline to following optimization steps which would then strive to get as close to this ideal solution as possible.

Step 2: Micro-channel distribution for the ideal case: Interlayer micro-channels are now placed such that: a) the heat levels decided by step 1 are effectively removed and the grid temperatures are as close to those prescribed by step 1 as possible, b) micro-channels are not allocated in areas with TSVs, and c) smallest number of channels are allocated for minimal pumping power.

Step 3: Gate size and grid temperature refinement: Since step 2 will be unable to entirely meet the ideal case solution of step 1, the gate size and grid temperature solution needs to be refined to account for the current micro-channel network in place.

Step 4: Micro-channel distribution refinement: The solution from step 3 gives a modified gate size and grid temperature prescription. Hence the micro-channel network needs to be refined further.

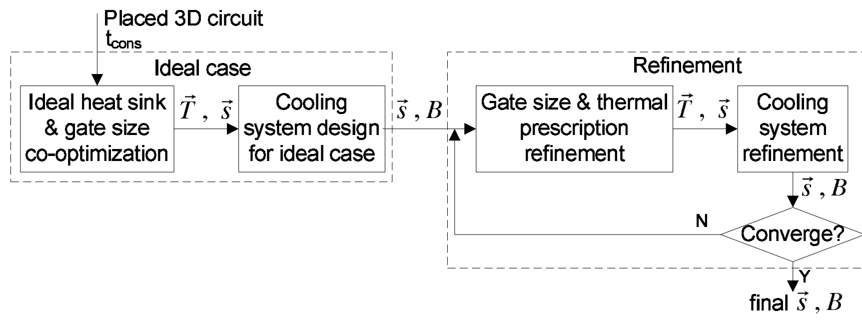


Figure 4.9: Overall design flow

Step 5: Iterate steps 3 and 4 till convergence criteria is met: The convergence criteria could be set to a maximum number of iterations or levels of improvements achieved.

Figure 4.9 illustrates the overall approach. In each step we strive to use algorithms and heuristics which draw upon rigorous optimization theory while exploiting the structure in the problem formulation. Now we describe each step in detail.

4.3.4.1 Step 1: Ideal Heat Sink and Gate Size Co-optimization

Let us first simplify the optimization problem in Equation 4.5 as:

$$\begin{aligned}
&\textbf{Decision variables}: \vec{s},\ \vec{T} \\
&min \sum_{\forall gate:g_i} (P_{d,i}(s_i) + P_{l,i}(s_i, T_i)) + \lambda \sum_{\forall grid:i} \frac{1}{T_i} \\
&s.t.\ \ 1.\quad t_j + d_i(\vec{s}, T_i) \le t_i, \forall \text{gate } g_i, g_j \in FI(g_i) \\
&\qquad 2.\quad t_i < t_{con}, \forall \text{gate } g_i \in PO \\
&\qquad 3.\quad 0 \le \vec{T} \le \vec{T}_{max} \\
&\qquad 4.\quad s_{min} \le s_i \le s_{max}, \forall \text{gate } g_i
\end{aligned} \tag{4.6}$$

In this formulation, the grid temperature T_i is assumed to be perfectly controllable through an ideal heat sink. The constraints signify meeting the timing constraint while staying with temperature and gate size constraints. The objective has two components: minimization of power as well as an additional term $\sum_{\forall grid:i} \frac{1}{T_i}$. This term signifies the fact that reducing T_i comes at the penalty of a more complex heat sink (which would be designed in the subsequent steps). Without this term, this optimization problem would trivially assign all T_i to be as small as possible (because that would benefit both timing and power). The solution of this problem represents allocation of gate sizes along with grid temperature, and would be used as a starting point for further optimization.

In order to solve this problem we make the following transformation $s_i = e^{x_i}$ and $T_i = e^{y_i}$. Based on this transformation, the gate delay and power consumption

models described in Sections 2.4.1 and 2.4.2 become: $d_i = e^{\sigma y_i} \cdot (\eta_{0i} + \sum_{\forall k \in FO(g_i)} \eta_{ki} \cdot e^{x_k - x_i})$, $P_{d,i} = \beta_{d,i} F e^{x_i}$, $P_{l,i} = e^{x_i} \cdot (\varepsilon_1 e^{2y_i} + \varepsilon_2 e^{y_i} + \varepsilon_3)$. It can be seen that the models for delay, leakage and dynamic powers are convex functions of variables x_i and y_i.

Theorem 1: Formulation is Equation 4.6 can be solved optimally using convex optimization approaches.

Proof: As indicated, gate delay, dynamic and leakage power functions are convex w.r.t. variables x_i and y_i. Hence the constraints are convex. The term $\sum_{\forall grid:i} \frac{1}{T_i}$ gets transformed to $\sum_{\forall grid:i} e^{-y_i}$ which is a convex function, too. Hence the overall objective function is convex as well, making the whole formulation optimally solvable using polynomial time convex methods. ■

4.3.4.2 Step 2: Micro-channel Distribution for Ideal Case

Step 1 has assigned gate sizes and grid temperature values. The gate sizes and temperatures decide the overall power dissipation profile while the temperature assignments indicate the level of cooling necessary in each grid. Together, these two aspects profoundly impact the design of the interlayer micro-fluidic system. The problem with the "ideal formulation" of step 1 is that it assumes perfect control of each grid temperature which is not possible even with interlayer micro-fluidics. By nature, micro-fluidic channels carry heat along the direction of fluid flow. They are incapable of controlling grid level temperatures. This is because, even though they enable localized cooling, they cannot completely remove the thermal cross-coupling of neighboring grids. The decision of allocating or removing a micro-channel will influence all the grids adjacent to this micro-channel. Hence in this step, we would

like to allocate channels such that the power dissipation levels are removed while ensuring the grid temperatures are as close as possible to the prescribed levels from step 1. We use least square fit (LSF) to find the micro-channel placement:

$$\min \quad \| \mathbf{G}(\mathbf{B}) \cdot \vec{T}_{desire} - \vec{P}_{desire} \|_2 \tag{4.7}$$

Here $\vec{T}_{desire}$ is the prescribed thermal profile decided by the previous step. $\vec{P}_{desire}$ is the sum of dynamic and leakage power calculated based on the prescribed gate sizes and temperatures using the power models in Sections 2.4.1 and 2.4.2. The objective is to decide the channel allocation such that the RMS (root-square-mean) error is minimized. $\mathbf{B}$ is the allocation of micro-channels and $\mathbf{G}(\mathbf{B})$ is the associated thermal conductivity matrix. For a given allocation of micro-channels, the associated conductivity matrix could be generated using the modeling approach described in Section 2.3. It is noteworthy that for a given set of potential channel locations, we would like to choose a subset such that the aforementioned objective is minimized.

To solve this, we first formulate the problem as an integer program. Essentially we assign a decision variable for each potential micro-channel location (binary constraint) and show that the conductivity matrix $\mathbf{G}$ is a linear function of these binary variables (proofs are omitted here). By approximating the binary variables as continuous, this problem becomes minimizing the RMS error of an affine function (since $\vec{T}_{desire}$ and $\vec{P}_{desire}$ are known, $(\mathbf{G}(\mathbf{B}) \cdot \vec{T}_{desire} - \vec{P}_{desire})$ is a linear function of $\mathbf{B}$), which can be solved efficiently. After solving this problem, we roundup the continuous variables to obtain the locations of micro-channels. Note that the objective

here is to generate a fluidic cooling solution that come as close as possible to the prescribed $\vec{T}_{desire}$ and $\vec{P}_{desire}$.

4.3.4.3 Step 3: Gate Size and Grid Temperature Refinement

Since the micro-channel solution from step 2 may not be able to come very close to the solution desired by step 1, we need to refine the original solution. Following are the objectives of this refinement step. 1) Step 2 synthesized a micro-channel solution which controls how power and temperature impact each other. This needs to be accounted for in the gate sizing solution. The ideal case of step 1 had assumed a perfectly controllable grid temperature. With the new channel infrastructure in-place, this assumption does not hold anymore. Hence the gate sizing needs to be re-evaluated. 2) We may still want to refine the channel structure further, based on newly prescribed temperature and gate sizes. Hence we would like to generate new assignments for grid temperature while accounting for the current cooling system in place.

In order to achieve the latter objective we divide the temperature T_i into two components: controllable and uncontrollable parts, $T_{c,i}$ and $T_{nc,i}$. The uncontrollable temperature is decided by the relationship between power and temperature which is a function of gate sizes and also the micro-channel structure in place. The controllable part is an additional parameter which we can control to prescribe any change in temperature. It would be used to further refine the micro-channel structure. The gate/grid temperature $T_i = T_{nc,i} \cdot T_{c,i}$. Here $T_{c,i} = 1$ indicates no change

at gate g_i (or grid i), $T_{c,i} < 1$ indicates greater need for cooling and $T_{c,i} > 1$ indicates less cooling necessary. The formulation at this step can be represented as follows.

$$\textbf{Decision variables}: \vec{s},\ \vec{T}_{nc},\ \vec{T}_c$$

$$Objective: \tag{4.8}$$

$$min \sum_{\forall gate:g_i} (P_{d,i}(s_i) + P_{l,i}(s_i, T_{nc,i} \cdot T_{c,i})) + \lambda \sum_{\forall grid:i} \frac{1}{T_{c,i}}$$

The objective structure is the same as the ideal case in step 1. However, the temperature affecting the gate leakage has two components now: uncontrollable part $T_{nc,i}$ and controllable part $T_{c,i}$. Because the controllable component is being assigned by us in this step, we would like $T_{c,i}$ to be as large as possible indicating minimal need for channels. This would help reduce pumping power. Hence the objective combines total power dissipated (the first two terms) along with pumping power (the third term).

$$Constraints\ 1,\ 2:$$

$$1.\quad t_j + d_i(\vec{s}, T_{nc,i} \cdot T_{c,i}) \leq t_i, \forall \text{gate } g_i, g_j \in FI(g_i) \tag{4.9}$$

$$2.\quad t_i < t_{con}, \forall \text{gate } g_i \in PO$$

This set of timing constraints (constraints 1 and 2) is similar to the ideal case except the gate temperature has two components.

$$Constraint\ 3: \quad \mathbf{G}(\mathbf{B}) \cdot \vec{T}_{nc} = \vec{P}_d(\vec{s}) + \vec{P}_l(\vec{s}, \vec{T}_{nc}) \tag{4.10}$$

As indicated earlier, $T_{nc,i}$ is the uncontrollable temperature which is decided by the power being dissipated and also the cooling system in place. Constraint 3 establishes the relationship between chip power dissipation and $T_{nc,i}$. Note that we do not include $T_{c,i}$ in this equation, because this parameter is being controlled to prescribe refinements in the cooling system, and would be used by future steps to redesign the cooling system.

Unlike the ideal case in step 1, $T_{c,i}$ should not be arbitrarily assigned in each grid since we already have a micro-channel network in place. For example, if a grid i already has a channel underneath, then increasing $T_{c,i}$ would prescribe removal of this channel. But doing so without accounting for the impact on other grids may result in significant sub-optimality since removal of a channel would affect a large number of grids. Also, if a grid i is located close to a TSV, then even if it has a small value of $T_{c,i}$ (indicating a need for channels), its extra cooling demands may never be met due to physical constraints imposed by TSVs. To account for these issues, the following constraints are imposes on the control of $T_{c,i}$.

$$Constraints\ 4,\ 5:$$

$$4.\quad \vec{T}_{c,min} \leq \vec{T}_c \leq \vec{T}_{c,max} \tag{4.11}$$

$$5.\quad T_{c,i} = T_{c,j}, \forall \text{adjacent grids } i, j \text{ along channel direction}$$

$T_{c,min,i}$ and $T_{c,max,i}$ values control how the $T_{c,i}$ values are allocated ($\vec{T}_{c,max}$, $\vec{T}_{c,min}$ are vectorized $T_{c,max,i}$, $T_{c,min,i}$). $T_{c,min,i} \leq 1$ and $T_{c,max,i} \geq 1$. A small value of $T_{c,min,i}$ implies the possibility of adding more cooling around grid i, while a large value of

$T_{c,min,i}$ implies smaller chance of adding extra cooling around i. Similarly, a large value of $T_{c,max,i}$ implies that grid i is close to some existing channels, hence great temperature increase would occur if the cooling around grid i is removed. A small value of $T_{c,max,i}$ implies that the impact of existing cooling configuration on grid i is small since they are far away. By appropriately assigning the values for $T_{c,min,i}$ and $T_{c,max,i}$, we can control the degree of change that is prescribed to the cooling system by the optimization formulation. The $T_{c,min,i}$ and $T_{c,max,i}$ values for each $T_{c,i}$ are allocated using the following rules.

Rule 1: If grid i is in the close vicinity of a TSV, then allocating channels nearby would be tougher. Hence we do not wish to have too much additional control of temperature at grid i. Therefore, $T_{c,min,i}$ and $T_{c,max,i}$ are allocated to be closer to each other such that significant changes in the fluidic structure around i is not prescribed by the optimization formulation. We use a formula based on distance and number of closeby TSVs to compute this range.

Rule 2: If a channel is already allocated very close to grid i, then $T_{c,min,i}$ is assigned to 1 and $T_{c,max,i}$ is assigned to be a large value. This indicates that the step 3 formulation only has the option of suggesting removal of a channel from this location.

Rule 3: If a channel is allocated close but not too close to a grid i, then $T_{c,min,i} < 1$ and its value is a function of the number of potential channel locations in the close vicinity. More the potential channel locations, smaller the value of $T_{c,min,i}$. $T_{c,max,i}$ is allocated to be a value greater than 1, and is a function of the distance to the closest channel in the *current* design. Greater the distance smaller

the value of $T_{c,max,i}$. This is because, prescribing an increase in grid temperature by removing channels will only be effective if they are located sufficiently close.

Rule 4: If no channel is allocated in sufficient vicinity then $T_{c,min,i}$ has the smallest value *possible* indicating that a channel could be added and $T_{c,max,i} = 1$ indicating that there is little possibility of removal of a channel.

Rule 5: All $T_{c,i}$ for the grids along the same micro-channel is allocated to be the same. Since each micro-channel spans the whole interlayer region in z direction, hence the prescribed changes for grids along the same micro-channel are assigned be the same due to the nature of micro-channels. This is illustrated in constraint 5.

Allocating $T_{c,min,i}$ and $T_{c,max,i}$ values is very critical since the ranges decide what kind of changes from the current fluidic structure end up being prescribed. The rules above attempt to constrain the formulation of step 3 to prescribe changes which are in sync with the current fluidic system in place. Also, as we re-iterate, we would like to make fewer modifications in the micro-channel structure. This could be achieved by reducing the range for $T_{c,i}$ as iterations progress.

Solving this formulation is more complex than the ideal case of step 1. Here too, we transform the temperature $T_{nc,i} = e^{y_{nc,i}}$, $T_{c,i} = e^{y_{c,i}}$, and gate size $s_i = e^{x_i}$. Hence the prescribed temperature $T_i = T_{nc,i} \cdot T_{c,i} = e^{y_{nc,i}+y_{c,i}}$. With this transformation, the gate delay, dynamic and leakage power become convex functions of the gate size and temperature variables $x_i, y_{nc,i}$ and $y_{c,i}$. The objective and constraints 1,2 in Equation 4.8, 4.9 remain convex. Constraints 4 and 5 are also convex (since ranges of the primary variables could be transformed to appropriate ranges of the transformed variables). Constraint 3, however is problematic. In this constraint,

$T_{nc,i}$ and power dissipation values are convex functions of x_i and $y_{nc,i}$. However the equality relationship in the constraint causes the convexity to breakdown. In order to address this problem, we represent the the power dissipation of gate g_i (leakage + dynamic) as a piecewise linear function of the gate size parameter x_i and uncontrollable temperature variable $y_{nc,i}$. Note that the right hand side of the constraint is basically the power dissipation for all gates. We also represent $T_{nc,i} = e^{y_{nc,i}}$ (on the left had side) as a piecewise linear function of $y_{nc,i}$. The underlying model parameters could be used to generate the coefficients for the piecewise linearization (these are standard approaches and therefore omitted for brevity). Because, both gate power dissipation and $T_{nc,i}$ are convex functions of x_i and $y_{nc,i}$, the following approach can be used to replace the variables $T_{nc,i}, P_{d,i}, P_{l,i}$ from constraint 3 by the underlying piecewise linearization.

$$\begin{aligned} &Power_i \geq \phi_{m,1} \cdot x_i + \phi_{m,2} \cdot y_{nc,i} + \phi_{m,3} \quad \forall m = 1...M \\ &Temp_i \geq \phi_{n,1} \cdot y_{nc,i} + \phi_{n,2} \quad \forall n = 1...N \end{aligned} \tag{4.12}$$

Here M and N are the number of linearizations imposed on the gate power dissipation and $T_{nc,i}$. Here $Power_i$ represents an upper bound on gate g_i's total power. The M-piecewise linearization is derived from the underlying model. Similarly $Temp_i$ is an upper bound on $T_{nc,i}$. Constraint 3 is now written as:

$$Constraint\ 3: \quad \mathbf{G}(\mathbf{B}) \cdot \vec{Temp} = \vec{Power} \tag{4.13}$$

Here $\vec{Temp}$ and $\vec{Power}$ are vectorized $Power_i$ and $Temp_i$. This modification enables

us to linearize constraint 3, which could now be augment with the other constraints and solved with standard convex optimization methods. The final solution of this optimization would be $x_i, y_{nc,i}$ and $y_{c,i}$ values for all gates. These would now be used to refine the micro-channel distribution.

4.3.4.4 Step 4: Micro-channel Distribution Refinement

Just as step 2, we would like to design the micro-channel distribution to address the heat dissipation decided by the gate sizes (and temperature) and also account for the change in the current configuration prescribed by $T_{c,i}$. This step is basically the same as step 2. However there are a few changes. Firstly, the formulation solved in step 3 uses upper bound $Power_i$ and $Temp_i$ as illustrated in Equations 4.12, 4.13. Hence, for a given gate size and micro-fluidic configuration, we will need to recompute the actual uncontrollable thermal profile $\vec{T}_{nc}$ (which could be done by simply solving Equation 4.10 for the assigned gate size). Note that this is a complex equation to solve due to leakage thermal interdependence. This would give the actual $\vec{T}_{nc}$ profile for the given gate size solution. Now we combine the actual $T_{nc,i}$ with the prescribed $T_{c,i}$ values to obtain the target grid temperature $T_i = T_{nc,i} \cdot T_{c,i}$. The generated target thermal profile is basically $\vec{T}_{desire}$ in step 2. Since the target thermal profile and gate sizes are known, the chip power profile could be computed as well. This would constitute $\vec{P}_{desire}$. Using these values, a new channel distribution is computed using techniques described in step 2.

4.3.4.5 Step 5: Re-iteration and Stopping Criteria

Steps 3, 4 are iterated to continue improvement in the overall solution. Firstly we would like to point out that the formulation in step 3, indirectly captures pumping power using the term $\lambda \sum_{\forall grid:i} \frac{1}{T_{c,i}}$. Secondly, as we iterate, Equation 4.11 controls the tolerable level of change from the current micro-channel allocation. By shrinking the range of $T_{c,i}$ as we iterate, the amount of change in the cooling solution becomes lesser and lesser. Hence after a few iterations, it will converge. This approach unifies the design of cooling structure with gate sizing. This is a significant improvement over conventional approaches that usually design the cooling infrastructure after designing the electrical aspects. In the next section we illustrate how such co-design can fundamentally improve the power-performance tradeoff in 3D-ICs.

4.3.5 Performance of Gate Sizing and Micro-channel Placement Co-design

To verify the power and performance improvement achieved by our approach, we compare our co-optimized design with two other approaches.

1. The thermal aware gate sizing approach with pure air cooling (*Air Cool* approach). In this approach, the overall thermal resistance of the heat sink for air cooling is 0.5°C/W.

2. The postfix approach that performs gate sizing first and then place micro-channel using the approach in [76] (*Postfix* approach).

The experimental setup is the similar as Section 3.7. In this experiment, we place a total of 2000 TSVs in the whitespace. The parameters of delay, thermal and power models are obtained from [47][86][91] and SPICE simulation.

4.3.5.1 Comparison of Power Consumption

We compare the total power consumption resulted from the three approaches. For the *Air Cool* approach, the power consumption consists of dynamic and leakage power, while for *Postfix* and our approaches, the total power consumption also includes the pumping power consumed by micro-channels. Table 4.4 shows the power consumption resulted from these approaches. For each benchmark, we tested power consumption for different timing constraints: one is tight and the other is looser. Note the *tight* timing constraint is the best achievable timing constraint for *Air Cool* approach (basically the tightest timing constraint that we can compare). Table 4.4 shows that, under the same performance constraint, our approach can result in 13.33% total power savings compared with *Air Cool* approach, indicating that the use of micro-channels, not only does not increase the system total power consumption, but actually helps save power instead. Compared with *Postfix* approach which performs gate sizing and micro-channel placement separately, our co-design approach achieves 12.05% power saving. This is because: a) micro-channel structure is optimized, b) micro-channels, which reduce chip temperature, also help reduce the leakage power and circuit delay, causing a favorable positive feedback.

Table 4.4: Comparison of total power consumption (power: W, t_{cons}: ns)

Bench mark	#Gates	t_{con} (tight/loose)	Total power			Power saving w.r.t	
			Air Cool	Postfix	Our	Air Cool	Postfix
1	343380	48 (tight)	294	289	254	13.61%	12.11%
		70 (loose)	226	223	197	12.83%	11.66%
2	394152	74 (tight)	256	251	219	14.45%	12.75%
		95 (loose)	233	219	189	18.88%	13.70%
3	342267	70 (tight)	221	218	191	13.57%	12.39%
		90 (loose)	182	189	164	9.89%	13.23%
4	295632	39 (tight)	293	287	258	11.95%	10.10%
		60 (loose)	214	210	189	11.68%	10.00%
5	208575	51 (tight)	284	291	248	12.67%	14.78%
		61 (loose)	251	245	219	12.75%	10.61%
6	181722	55 (tight)	232	232	206	11.21%	11.21%
		75 (loose)	190	188	167	12.11%	11.17%
Average			240	237	208	13.33%	12.05%

4.3.5.2 Comparison of Circuit Delay

We also compare the best achievable circuit delay under the same power envelop. This was obtained by performing a binary search on timing constraints t_{con}. Table 4.5 shows that our co-optimized design achieves 15.88% circuit speedup over the *Air cool* and *Postfix* approaches, while still consuming the same (or even less) amount of power.

4.3.6 Power-Performance Tradeoff

To characterize the tradeoff between the system performance and power consumption, we plot the circuit delay versus power consumption for benchmark 1 as Figure 4.10 shows. For all three approaches, the power consumption increases as the

Table 4.5: Comparison of circuit performance (power: W, t_{cons}: ns)

Bench mark	Air cool		Postfix		Our		Circuit speedup
	Best t_{con}	Power	Best t_{con}	Power	Best t_{con}	Power	
1	48	294	48	289	40	289	16.67%
2	74	256	74	251	60	251	18.92%
3	70	221	70	218	57	218	18.57%
4	39	293	39	287	34	287	12.82%
5	51	284	51	291	44	277	13.73%
6	55	232	55	232	47	231	14.55%
Average	56	263	56	261	47	259	15.88%

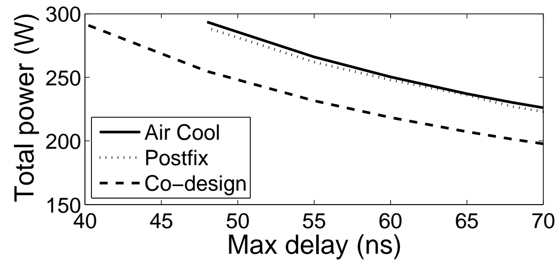


Figure 4.10: Delay versus power tradeoff for benchmark 1

timing constraint becomes tighter. In the figure, the solid line is the power consumption of conventional gate sizing approach using pure air cooling. This line is basically the best power-delay tradeoff that the conventional gate sizing approach can achieve. The tradeoff achieved by *Postfix* approach has slight (but not significant) improvement over the conventional gate sizing approach. However, using co-design results in significant performance-power improvement. The figure shows that for all timing constraints we tested, our design always dissipates less power compared with the other two approaches. Similarly, when the available power budget is fixed, our design achieves better circuit speed, indicating a fundamental power-performance improvement achieved by 3D-IC electric and cooling system co-design.

4.4 Summary

In this chapter, we investigated two electrical-cooling system co-design problems: a) TSV assignment and micro-fluidic cooling co-optimization, and b) gate sizing and micro-fluidic cooling co-optimization.

We firstly investigated a co-optimization of TSV assignment to interlayer nets and micro-channel allocation such that both wirelength and micro-channel cooling energy are co-optimized. We propose a multi-commodity min-cost flow based formulation followed by simplifying transformations that enable use of effective polynomial time heuristics. The experimental results show that, our co-optimization approach achieves 46% cooling power savings or 7.6% wire length reduction compared with the approaches that assign TSVs and allocate micro-channels separately.

We then investigated a co-optimization approach for 3D-IC gate sizing and micro-fluidic cooling design that fully exploits the interdependency between power, temperature and circuit delay to push the power-performance tradeoff beyond conventional limits. We proposed a unified formulation to model this co-optimization problem and use an iterative optimization approach to solve the problem. The experimental results show a fundamental power-performance improvement, with 12% power saving and 16% circuit speedup.

Compared with the conventional design flow that separates the electrical and cooling system design, the co-design methodology can fundamentally improve the system power and performance. Furthermore, it also allows a more flexible tradeoff between the system performance (such as wirelength and circuit delay) and power

consumption.

Chapter 5

Conclusion and Discussion

5.1 Conclusion

In this work, we investigated several aspects of micro-fluidic cooling for 3D-ICs. The micro-fluidic cooling is capable of removing very high density heat. However, there are also overhead or constraints associated micro-fluidic cooling, such as significant extra cooling power consumption, resource conflict with TSVs, etc.

In order to overcome these overheads or account for the design constraints, we proposed three micro-fluidic cooling configurations that can result in significant cooling power savings and meanwhile, avoid the TSVs. In these designs, micro-channel structures are designed after the electrical part of the chip, hence they are compatible with the standard IC design flow. Besides optimized cooling configuration, we also proposed a micro-channel based dynamic thermal management method that controls the fluid velocity at runtime to allow real time thermal control.

The electrical, thermal, reliability and cooling aspects are all interdependent. Therefore, although these cooling system designs are compatible with the standard IC design flow, separating the design of electrical and cooling system actually leads to sub-optimal designs. Hence, we then investigated the electrical and cooling system co-design to achieve further power-performance improvement.

We firstly investigated a co-optimization of TSV assignment to interlayer nets

and micro-channel allocation such that both wirelength and micro-channel cooling energy are co-optimized. We propose a multi-commodity flow based formulation followed by simplifying transformations that enable use of effective polynomial time heuristics. The experimental results show that, our co-optimization approach achieves 46% cooling power savings or 7.6% wire length reduction compared with the approaches that assign TSVs and allocate micro-channels separately.

We then investigated a co-optimization approach for 3D-IC gate sizing and micro-fluidic cooling design that fully exploits the interdependency between power, temperature and circuit delay to push the power-performance tradeoff beyond conventional limits. We proposed a unified formulation to model this co-optimization problem and use an iterative optimization approach to solve the problem. The experimental results show a fundamental power-performance improvement, with 12% power saving and 16% circuit speedup.

With the existence of micro-fluidic cooling, the designers now can perform a more aggressive performance optimization, since the resulting heat can be removed by the liquid flow in the micro-channels. Furthermore, the co-optimization will help us fully exploit the advantages of micro-fluidic cooling and result in a fundamental improvement in the system power-performance tradeoff.

5.2 Future Work

Using of micro-fluidic cooling in the 3D-IC is still a new technology and several problems need to be addressed.

The first direction is more extensive investigation of electro-thermo-mechanical co-design. The existence of micro-channels not only influences the gate sizing and TSV assignment as explored in Chapter 4, it will change the whole physical design process, such as 3D-IC partitioning and floorplanning etc. For example, the 3D-IC partitioning can be optimized more aggressively to achieve better bandwidth; the floorplanning can also be optimized to save chip area, etc. Besides physical design, the micro-fluidic cooling also enables a more aggressive architectural level design without worrying about the temperature, since the resulting heat can be removed by micro-channels.

The second direction is reliability associated with micro-channels. As mentioned earlier, in the 3D-IC, TSVs are incorporated to enable interlayer communications and delivery of power/ground. Copper, due to its low resistivity, is a commonly used material for TSV fill. Since the chips are usually annealed at the temperature level much higher than their operating temperature, when cooling down from the annealing temperature, thermal stress occurs due to the coefficient of thermal expansion (CTE) mismatch between the TSV fill material (e.g. copper) and silicon. The thermal stress might cause reliability problems such as cracking. The existence of micro-channels will change the 3D-IC thermal profile and hence influence the thermal stress field inside 3D-ICs as well. The impact of micro-fluidic cooling on chip reliability (through thermal stress) needs to be analyzed. Besides thermal stress, the coolant fluid inside micro-channels also causes mechanical stress on micro-channel sidewalls. This intensity of such stress depends on the distribution, dimension of micro-channels and fluid flow rate (velocity) through micro-channels (along with

choice of material). Such mechanical stress also needs to be investigated.

Furthermore, the thermal stress inside 3D-IC also influences the carrier mobilities, hence affecting gate delays. The impact of stress on gate/circuit delay is complex, depending on the intensity of stress, the location of gates and TSVs and the type of transistor (NMOS or PMOS). The micro-fluidic cooling, since it influences the thermal stress, also influences the circuit delay through thermal stress. As a result, it will fundamentally change the timing analysis in 3D-ICs. When performing statistical timing analysis in 3D-IC, we should take this fact into consideration [74]. Moreover, in designing the micro-fluidic cooling configurations, this thermal stress effect should also be considered, which basically requires electrical and cooling system co-design as well.

Bibliography

[1] Capo: a large-scale fixed-die floorplacer. *http://vlsicad.eecs.umich.edu/BK/PDtools/Capo/.*

[2] Ibm-place 2.0 benchmark. In *http://er.cs.ucla.edu/benchmarks/ibm-place2/.*

[3] Labyrinth global router. In *http://cseweb.ucsd.edu/ kastner/research/labyrinth/.*

[4] ITC'99 benchmarks. *http://www.cad.polito.it/dow nloads/tools/itc99.html.*

[5] T. M. Adams, S. I. Abdel-Khalik, S. M. Jeter, and Z. H. Qureshi. An experimental investigation of single-phase forced convection in microchannels. *International Journal of Heat and Mass Transfer*, pages 851–857, 1998.

[6] Bruno Agostini, John Richard Thome, Matteo Fabbri, and Bruno Michel. High heat flux two-phase cooling in silicon multimicrochannels. *IEEE Transactions on Components and Packaging Technologies, Vol.31*, 2008.

[7] K. Athikulwongse, A. Chakraborty, Jae-Seok Yang, D.Z. Pan, and Sung Kyu Lim. Stress-driven 3d-ic placement with tsv keep-out zone and regularity study. In *IEEE/ACM Intl. Conf. on Computer Aided Design (ICCAD'10)*, 2010.

[8] Muhannad S. Bakir, Calvin King, and et al. 3D heterogeneous integrated systems: Liquid cooling, power delivery, and implementation. In *IEEE Custom Intergrated Circuits Conference*, pages 663–670, 2008.

[9] Avram Bar-Cohen. Thermal management of on-chip hot spots and 3d chip stacks. In *IEEE International Conference on Microwaves, Communications, Antennas and Electronics Systems*, pages 1–8, 2009.

[10] James R Black. Electromigrationa brief survey and some recent results. *IEEE Transactions on Electron Devices*, 16:338–347, 1969.

[11] David Brooks and Margaret Martonosi. Dynamic thermal management for high-performance microprocessors. In *Proc. of the 7th Intl. Symp. on High-Performance Computer Architecture (HPCA'01).*

[12] Thomas Brunschwiler, Bruno Michel, Hugo Rothuizen, Urs Kloter, Bernhard Wunderle, and Herbert Reichl. Hotspot-optimized interlayer cooling in vertically integrated packages. *Proc. Materials Research Society (MRS) Fall Meeting*, 2008.

[13] Thomas D Burd, Trevor A Pering, Anthony J Stratakos, and Robert W Brodersen. A dynamic voltage scaled microprocessor system. *Solid-State Circuits, IEEE Journal of*, 35:1571–1580, 2000.

[14] Ting-Yen Chiang, K. Banerjee, and K.C. Saraswat. Effect of via separation and low-k dielectric materials on the thermal characteristics of Cu interconnects. In *IEEE Intl. Electron Devices Meeting, IEDM Technical Digest*, pages 261–264, 2000.

[15] S.B. Choi, R.F. Barron, and R.O. Warrington. Fluid flow and heat transfer in micro tubes. *Micromechanical sensors, actuators and systems, ASME DSC*, pages 123–128, 1991.

[16] Aviad Cohen, Lev Finkelstein, Avi Mendelson, Ronny Ronen, and Dmitry Rudoy. On estimating optimal performance of cpu dynamic thermal management. *IEEE Computer Architecture Letters*, 2:6, 2003.

[17] Jason Cong and Yan Zhang. Thermal via planning for 3-D ICs. In *IEEE/ACM Intl. Conf. on Computer Aided Design (ICCAD'05)*, pages 744–751, 2005.

[18] Ayse K. Coskun, David Atienza, Tajana Simunic Rosing, and et al. Energy-efficient variable-flow liquid cooling in 3D stacked architectures. In *Conference on Design, Automation and Test in Europe (DATE'10)*, pages 111–116, 2010.

[19] Ayse K. Coskun, Jose L. Ayala, David Atienzaz, and Tajana Simunic Rosing. Modeling and dynamic management of 3D multicore systems with liquid cooling. In *17th Annual IFIP/IEEE International Conference on Very Large Scale Integration*, pages 60–65, 2009.

[20] Ayse Kivilcim Coskun, Tajana Simunic Rosing, and Kenny C. Gross. Temperature management in microprocessor socs using online learning. In *Design Automation Conference (DAC'08)*.

[21] Ayse Kivilcim Coskun, Tajana Simunic Rosing, and Kenny C. Gross. Proactive temperature management in MPSoCs. In *Proceedings of the 2008 International Symposium on Low Power Electronics and Design*, pages 165–170, 2008.

[22] William J Dally. Future directions for on-chip interconnection networks. In *OCIN Workshop*, 2006.

[23] Lotfollah Ghodoossi. Thermal and hydrodynamic analysis of a fractal microchannel network. *Energy Conversion and Management, Elsevier*, pages 771–788, 2005.

[24] Brent Goplen and Sachin Sapatnekar. Thermal via placement in 3D ICs. In *International Symposium on Physical Design (ISPD'05)*, pages 167–174, 2005.

[25] Vinay Hanumaiah, Sarma Vrudhula, and Karam S Chatha. Performance optimal online dvfs and task migration techniques for thermally constrained multicore processors. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 30:1677–1690, 2011.

[26] Michael B. Healy and Sung Kyu Lim. Power delivery system architecture for many-tier 3d systems. In *Electronic Components and Technology Conference*, pages 1682–1688, 2010.

[27] Huang Huang, Gang Quan, and Jeffrey Fan. Leakage temperature dependency modeling in system level analysis. In *11th International Symposium on Quality Electronic Design (ISQED)*, pages 447–452, 2010.

[28] H. Irie, K. Kita, K. Kyuno, and A. Toriumi. In-plane mobility anisotropy and universality under uni-axial strains in n- and p-mos inversion layers on (100), (110), and (111) si. In *IEEE International Electron Devices Meeting*, pages 225–228, 2004.

[29] Muhamad Amri Ismail, Iskhandar Md Nasir, and Razali Ismail. Modeling of temperature variations in mosfet mismatch for circuit simulations. In *Quality Electronic Design, 2009. ASQED 2009. 1st Asia Symposium on*, pages 357–362, 2009.

[30] Philip Jacob, Okan Erdogan, Aamir Zia, Paul M Belemjian, Russell P Kraft, and John F McDonald. Predicting the performance of a 3d processor-memory chip stack. *IEEE Design & Test of Computers*, 22:540–547, 2005.

[31] Arun Jagota and Laura A. Sanchis. Adaptive, restart, randomized greedy heuristics for maximum clique. *Journal of Heuristics*, 7:565 – 584, 2001.

[32] Linan Jiang, Jae-Mo Koo, and et al. Cross-linked microchannels for vlsi hotspot cooling. In *ASME 2002 International Mechanical Engineering Congress and Exposition*, 2002.

[33] Satish Kandlikar, Srinivas Garimella, and et al. Heat transfer and fluid flow in minichannels and microchannels. *Elsevier*, 2005.

[34] J. Keslin. Viscosity of liquid water in the range - 8 c to 150 c. *J. Phys. Chem. Ref. Data*, 7, 1978.

[35] Mahesh Ketkar, Kishore Kasamsetty, and Sachin S. Sapatnekar. Convex delay models for transistor sizing. In *Design Automation Conference (DAC'00)*, pages 655–660, 2000.

[36] Dae Hyun Kim, Krit Athikulwongse, and Sung Kyu Lim. A study of through-silicon-via impact on the 3D stacked IC layout. In *IEEE/ACM Intl. Conf. on Computer Aided Design (ICCAD'09)*, pages 674–680, 2009.

[37] Duckjong Kim, Sung Jin Kim, and Alfonso Ortega. Compact modeling of fluid flow and heat transfer in pin fin heat sinks. *Journal of Electronic Packaging*, 2004.

[38] N.S. Kim, T. Austin, D. Blaauw, T. Mudge, K. Flautner, J.S. Hu, M.J. Irwin, M. Kandemir, and V. Narayanan. Leakage current: Moores law meets static power. *IEEE Computer Society*, 36(12):68–75.

[39] Yoon Jo Kim, Yogendra K. Joshi, and et al. Thermal characterization of interlayer microfluidic cooling of three dimensional integrated circuits with nonuniform heat flux. *ASME Trans. Journel of Heat Transfer*, 2010.

[40] CR King, D. Sekar, M.S. Bakir, B. Dang, J. Pikarsky, and J.D. Meindl. 3d stacking of chips with electrical and microfluidic i/o interconnects. In *Electronic Components and Technology Conference*, pages 1–7, 2008.

[41] Alexander Klaiber et al. The technology behind crusoe processors. *Transmeta Technical Brief*, 2000.

[42] Roy W. Knight, Donald J. Hall, and et al. Heat sink optimization with application to microchannels. *IEEE Trans. on Components, Hybrids, and Manufacturing Technology*, pages 832–842, 1992.

[43] Jae-Mo Koo, Sungjun Im, Linan Jiang, and Kenneth E. Goodson. Integrated microchannel cooling for three-dimensional electronic circuit architectures. *ASME Trans. Journel of Heat Transfer*, pages 49–58, 2005.

[44] K. Laker and W. Sansen. Design of analog integrated circuits and systems. *New York: McGraw-Hill*, 1994.

[45] Young-Joon Lee and Sung Kyu Lim. Co-optimization of signal, power, and thermal distribution networks for 3D ICs. In *Electrical Design of Advanced Packaging and Systems Symposium*, pages 163–155, 2008.

[46] Jing Li and H. Miyashita. Efficient thermal via planning for placement of 3d integrated circuits. In *IEEE International Symposium on Circuits and Systems (ISCAS'07)*, pages 145–148, 2007.

[47] W. Liao, L. He, and K.M. Lepak. Temperature and supply voltage aware performance and power modeling at microarchitecture level. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Syst.*, 24:1042–1053, 2005.

[48] Xiaodong Liu, Yufan Zhang, Gary Yeap, and Xuan Zeng. An integrated algorithm for 3D-IC TSV assignment. In *DAC*, pages 652–657, 2011.

[49] James J-Q Lu, Ken Rose, and Susan Vitkavage. 3d integration: Why, what, who, when? *Future Fab Intl*, 23, 2007.

[50] Zhijian Lu, John Lach, Mircea Stan, and Kevin Skadron. Banking chip lifetime: Opportunities and implementation. In *Proceedings of the 1st Workshop on High Performance Computing Reliability Issues (HPCRI05)*, 2005.

[51] I. Hassan M. Dang and R. Muwanga. Adiabatic two phase flow distribution and visualization in scaled microchannel heat sinks. *Experiments in Fluids*, 2007.

[52] Christophe Marques and Kevin W. Kelly. Fabrication and performance of a pin fin micro heat exchanger. *Journal of Heat Transfer*, pages 434–444, 2004.

[53] Steven M. Martin, Krisztian Flautner, Trevor Mudge, and David Blaauw. Combined dynamic voltage scaling and adaptive body biasing for lower power microprocessors under dynamic workloads. In *IEEE/ACM Intl. Conf. on Computer Aided Design (ICCAD'02).*

[54] H. Mizunuma, C. L. Yang, and Y. C. Lu. Thermal modeling for 3D-ICs with integrated microchannel cooling. In *IEEE/ACM Intl. Conf. on Computer Aided Design*, pages 256–263, 2009.

[55] Bruce Roy Munson, Donald F. Young, Theodore H. Okiishi, and Wade W. Huebsch. Fundamentals of fluid mechanics. *Wiley*, 2008.

[56] Y. S. Muzychka and M. M. Yovanovich. Modelling friction factors in non-circular ducts for developing laminar flow. In *2nd AIAA Theoretical Fluid Mechanics Meeting*, 1998.

[57] Mohit Pathak, Young-Joon Lee, Thomas Moon, and Sung Kyu Lim. Through-silicon-via management during 3d physical design: When to add and how many? In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD'10)*, pages 387–394, 2010.

[58] Massoud Pedram and Shahin Nazarian. Thermal modeling, analysis and management in VLSI circuits: Principles and methods. *Proceedings of the IEEE*, 94:1487–1501, 2006.

[59] Yoav Peles, Ali Kosar, Chandan Mishra, Chih-Jung Kuo, and Brandon Schneider. Forced convective heat transfer across a pin fin micro heaet sink. *International Journal of Heat and Mass Transfer*, pages 3615–3627, 2005.

[60] Kiran Puttaswamy and Gabriel H. Loh. Thermal analysis of a 3D die-stacked high-performance microprocessor. In *Proceedings of the 16th ACM Great Lakes symposium on VLSI (GLSVLSI'06)*, 2006.

[61] Hanhua Qian, Xiwei Huang, Hao Yu, and Chip Hong Chang. Cyber-physical thermal management of 3d multi-core cache professor system with microfluidic cooling. *Journal of Low Power Electronics*, 2011.

[62] Weilin Qu, Issam Mudawar, Sang-Youp Lee, and Steven T. Wereley. Experimental and computational investigation of flow development and pressure drop in a rectangular micro-channel. *Journal of Electronic Packaging*, 2006.

[63] Ravishankar Rao and Sarma Vrudhula. Performance optimal processor throttling under thermal constraints. In *Proc. of Intl. Conf. on Compilers Architectures and Synthesis for Embedded Systems (CASES'07)*, pages 257–266, 2007.

[64] Ravishankar Rao, Sarma Vrudhula, and Naehyuck Chang. An optimal analytical solution for processor speed control with thermal constraints. In *Proceedings of the 2006 International Symposium on Low Power Electronics and Design*, pages 292–297, 2006.

[65] K. Ahuja Ravindra, L. Magnanti Thomas, and James B. Orlin. Network flows: Theory, algorithms and applications. *Prentice Hall*, 1993.

[66] Takashi Sato, Junji Ichimiya, Nobuto Ono, and Masanori Hashimoto. On-chip thermal gradient analysis considering interdependence between leakage power and temperature. In *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, pages 3491–3499, 2006.

[67] TH Schubert, Ł Ciupiński, J Morgiel, H Weidmüller, T Weissgärber, and B Kieback. Advanced composite materials for heat sink applications. *Euro PM*, 2007.

[68] S.M. Senn and D. Poulikakos. Laminar mixing, heat transfer and pressure drop in tree-like microchannel nets and their application for thermal management in polymer electrolyte fuel cells. *Journal of Power Sources, Vol. 130*, pages 178–191, 2004.

[69] R. K. Shah and A. L. London. Laminar flow forced convection in ducts: A source book for compact heat exchanger analytical data. *Academic*, 1978.

[70] Bing Shi, Caleb Serafy, and Ankur Srivastava. Co-optimization of tsv assignment and micro-channel placement for 3d-ics. In *Proceedings of the 23rd ACM international conference on Great lakes symposium on VLSI*, pages 337–338, 2013.

[71] Bing Shi and Ankur Srivastava. Cooling of 3d-ic using non-uniform micro-channels and sensor based dynamic thermal management. In *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, pages 1400–1407, 2011.

[72] Bing Shi and Ankur Srivastava. Liquid cooling for 3D-ICs. In *invited paper, First International IEEE Workshop on Thermal Modeling and Management: Chips to Data Centers*, 2011.

[73] Bing Shi and Ankur Srivastava. Tsv-constrained micro-channel infrastructure design for cooling stacked 3d-ics. In *Proceedings of the 2012 ACM international symposium on International Symposium on Physical Design*, pages 113–118, 2012.

[74] Bing Shi and Ankur Srivastava. Thermal stress aware 3d-ic statistical static timing analysis. In *Proceedings of the 23rd ACM international conference on Great lakes symposium on VLSI*, pages 281–286, 2013.

[75] Bing Shi, Ankur Srivastava, and Avram Bar-Cohen. Hybrid 3d-ic cooling system using micro-fluidic cooling and thermal tsvs. In *VLSI (ISVLSI), 2012 IEEE Computer Society Annual Symposium on*, pages 33–38, 2012.

[76] Bing Shi, Ankur Srivastava, and Peng Wang. Non-uniform micro-channel design for stacked 3D-ICs. In *Design Automation Conference (DAC'11)*, 2011.

[77] Kevin Skadron, Mircea R. Stan, Karthik Sankaranarayanan, Wei Huang, Sivakumar Velusamy, and David Tarjan. Temperature-aware microarchitecture: Modeling and implementation. *ACM Trans. on Architecture and Code Optimization*, 1(1):94–125, 3.

[78] Arvind Sridhar, Alessandro Vincenzi, Martino Ruggiero, Thomas Brunschwiler, and David Atienza. 3D-ICE: Fast compact transient thermal modeling for 3D ICs with inter-tier liquid cooling. In *IEEE/ACM Intl. Conf. on Computer Aided Design (ICCAD'10)*, 2010.

[79] Linda Stappers, Yanli Yuan, and Jan Fransaer. Novel composite coatings for heat sink applications. *Journal of The Electrochemical Society*, 152:C457–C461, 2005.

[80] Haihua Su, Frank Liu, Anirudh Devgan, Emrah Acar, and Sani Nassif. Full chip leakage estimation considering power supply and temperature variations. In *Proceedings of the 2003 international symposium on Low power electronics and design*, pages 78–83, 2003.

[81] Haihua Su, Frank Liu, Anirudh Devgan, Emrah Acar, and Sani Nassif. Full chip leakage estimation considering power supply and temperature variations. In *Proceedings of the 2003 International Symposium on Low Power Electronics and Design (ISLPED'03)*, pages 78 – 83, 2003.

[82] John R. Thome. Engineering data book iii. *Wolverine Tube*, 2004.

[83] Y. Tsividis. *Operation and Modeling of the Mos Transistor*. Oxford University Press, 2004.

[84] D. B. Tuckerman and R. F. W. Pease. High-performance heat sinking for VLSI. *IEEE Electron Device Letters*, pages 126–129, 1981.

[85] R. Walchli, T. Brunschwiler, B. Michel, and D. Poulikakos. Combined local microchannel-scale cfd modeling and global chip scale network modeling for electronics cooling design. *International Journal of Heat and Mass Transfer*, 2010.

[86] Neil Weste and David Harris. Cmos vlsi design: A circuits and systems perspective. *Addison Wesley*, 2010.

[87] Frank M. White. Fluid mechanics. *McGraw-Hill Book Company*, 1986.

[88] Dong Hyuk Woo, Nak Hee Seong, Dean L Lewis, and H-HS Lee. An optimized 3d-stacked memory architecture by exploiting excessive, high-density tsv bandwidth. In *IEEE 16th International Symposium on High Performance Computer Architecture (HPCA)*, pages 1–12, 2010.

[89] P. Y. Wu and W.A. Little. Measuring of the heat transfer characteristics of gas flow in fine channel heat exchangers for micro miniature refrigerators. *Cryogenics*, 1994.

[90] Jin-Tai Yan, Yu-Cheng Chang, and Zhi-Wei Chen. Thermal via planning for temperature reduction in 3D ICs. In *IEEE International SOC Conference (SOCC'10)*, pages 392–395, 2010.

[91] C.Y. Yang, J.J. Chen, L. Thiele, and T.W. Kuo. Energy-efficient real-time task scheduling with temperature-dependent leakage. In *Conference on Design, Automation and Test in Europe*, pages 9–14, 2010.

[92] Jae-Seok Yang, Krit Athikulwongse, Young-Joon Lee, Sung Kyu Lim, and David Z. Pan. Tsv stress aware timing analysis with applications to 3d-ic layout optimization. In *Proceedings of the 47th Design Automation Conference (DAC'10)*, 2010.

[93] Jun Yang, Xiuyi Zhou, Marek Chrobak, Youtao Zhang, and Lingling Jin. Dynamic thermal management through task scheduling. In *Performance Analysis of Systems and software, 2008. ISPASS 2008. IEEE International Symposium on*, pages 191–201, 2008.

[94] D. Yu, R. Warrington, R. Barron, and T. Ameen. An experimental and theoretical investigation of fluid flow and heat transfer in microtubes. *Proceedings of the ASME/JSME Thermal Engineering Conference*, pages 523–530, 1995.

[95] Tianpei Zhang, Yong Zhan, and Sachin S. Sapatnekar. Temperature-aware routing in 3D ICs. In *ASP-DAC*, pages 309–314, 2006.

[96] Yufu Zhang and Ankur Srivastava. Adaptive and autonomous thermal tracking for high performance computing systems. In *Design Automation Conference (DAC'10)*, 2010.