THE INSTITUTE FOR SYSTEMS RESEARCH

ISR TECHNICAL REPORT 2011-03

TSV-Constrained Micro-Channel Infrastructure Design for Cooling Stacked 3D-ICs

Bing Shi
Ankur Srivastava

The Institute for Systems Research

A. JAMES CLARK SCHOOL OF ENGINEERING

ISR develops, applies and teaches advanced methodologies of design and analysis to solve complex, hierarchical, heterogeneous and dynamic problems of engineering technology and systems for industry and government.

ISR is a permanent institute of the University of Maryland, within the A. James Clark School of Engineering. It is a graduated National Science Foundation Engineering Research Center.

www.isr.umd.edu

TSV-Constrained Micro-Channel Infrastructure Design for Cooling Stacked 3D-ICs

Bing Shi and Ankur Srivastava
University of Maryland, College Park, MD, USA
email: {bingshi, ankurs}@umd.edu

Abstract

Micro-channel based liquid cooling has significant capability of removing high density heat in 3D-ICs. The conventional micro-channel structures investigated for cooling 3D-ICs use straight channels. However, the presence of TSVs which form obstacles to the micro-channels prevents distribution of straight micro-channels. In this paper, we investigate the methodology of designing TSV-constrained micro-channel infrastructure. Specifically, we decide the locations and geometry of micro-channels with bended structure so that it's cooling effectiveness is maximized. Our micro-channel structure could achieve up to 87% pumping power saving compared with the micro-channel structure using straight channels.

1 Introduction and motivation

The three-dimensional integrated circuits (3D-IC) consists of two or more layers of active electronic components which are stacked vertically. Despite its significant performance improvement over 2D circuits such as fast on-chip communications, 3D-IC also exhibits thermal issues due to its high power density cause by the stacked architecture.

While the conventional air cooling might be not enough for stacked 3D-ICs [1], the micro-channel based liquid cooling provides a better option to address this problem. Micro-channels have significant capability of cooling high heat density (as much as $700W/cm^2$ [2]) and therefore are very appropriate for cooling 3D-ICs. Figure 1 shows a three-tier stacked 3D-IC integrated with micro-channel heat sinks [3] [4]. In each tier of the 3D-IC, the active layer (silicon layer) consists of functional units such as cores and memories which dissipates power. Micro-channel heat sinks are embedded below each silicon layer to provide cooling. The coolant fluid is pumped through the micro-channels, and takes away the heat generated in the silicon layer.

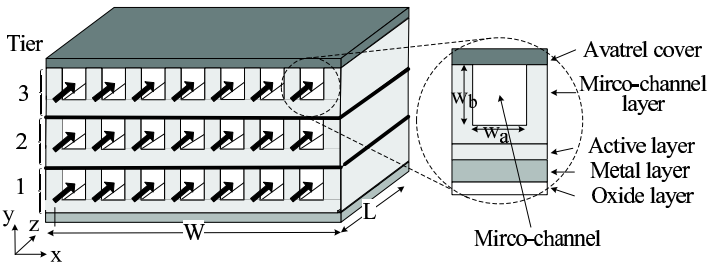


Figure 1: A 3D-IC integrated with micro-channels

Many works have investigated the thermal modeling of 3D-ICs with micro-channels heat sinks [1] [3] [4], including the study on thermal wake effect [5]. Some other works try to find the best dimensional parameters such as channel width, height and sidewall thickness so as to improve the overall cooling effectiveness of the micro-channel system [2] [6]. Micro-channel cooling is now also adopted in dynamic thermal management (DTM) to control the runtime CPU performance and chip temperature by tuning the fluid flow rate through micro-channels [7].

The conventional micro-channel structures investigated for cooling 3D-ICs use straight channels that spread on the whole chip or in areas that demand high cooling capacity. If the spatial distribution of micro-channels is unconstrained then such an approach results in the best cooling efficiency with the minimum cooling energy (power dissipated to pump the fluid). 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 [8] [9]. 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 prevents distribution of straight micro-channels. This results in the following problems.

1. As illustrated in figure 2(a), micro-channels would fail to reach thermally critical areas thereby resulting in thermal violations and hotspots.
2. In order to fix the thermal hotspots in areas where micro-channels cannot reach, we would need to increase the fluid flow rate resulting in a significant increase in cooling energy.

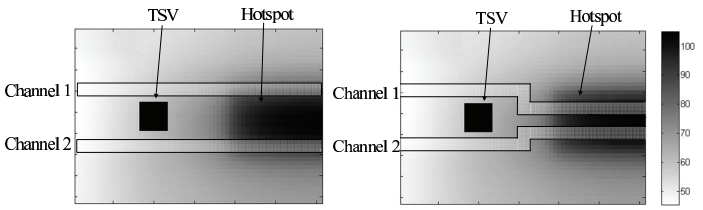


Figure 2: Example of silicon layer thermal profile with TSV and (a) straight, (b) bended micro-channels

To address this problem, we investigate micro-channel with bends as illustrated in figure 2(b). For ease of manufacturability, we only investigate channel structures that can have 90° bends. Such micro-channel structures could be easily manufactured in modern lithography technologies. With bended structure, the micro-channels can reach those TSV-blocked hotspot regions which straight micro-channels cannot reach. This results in better coverage of hotspots and therefore better cooling efficiency and reduced cooling energy. While micro-channels with bends (or serpentine organization of micro-channels) has been investigated in the past [10] [11], 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. Having bends in micro-channels impacts its cooling efficiency and generally should be avoided if an equivalent straight micro-channel structure can be incorporated. But in situations where presence of TSVs results in lack of hotspot coverage by straight micro-channels alone, bended structures would allow removal of excessive heat.

Designing 3D-IC micro-channel infrastructure with bends 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. In this paper, we investigate the methodology of designing TSV-constrained micro-channel infrastructure. Specifically, we decide the locations and geometry of micro-channels with bended structure so that it's cooling effectiveness is maximized (that is, sufficient cooling is achieved at minimum pumping power). Our micro-channel structure could achieve up to 87% pumping power saving compared with the micro-channel structure using straight channels.

The organization of this paper is as follows. In section 2, we introduce the thermal and power model of 3D-IC with micro-channel

cooling. We investigate the TSV-constrained micro-channel infrastructure design methodology in sections 3. The experimental result is given in section 4.

2 Thermal and power model for 3D-IC with micro-channels

2.1 Basic Heat Flow Mechanisms in Micro-channels

Consider the micro-channel section in figure 3, heat dissipated in the silicon layer *conducts* to the micro-channel sidewalls. The heat is then absorbed by the coolant fluid through *convection*. The heated fluid is then carried away by the moving flow. Therefore, the cooling effectiveness of a micro-channel contains three aspects: conduction, convection and fluid flow. When the heat dissipated by the system is non-uniform, the heated fluid from upstream (which has been heated by high power dissipation upstream) might result in the heating of cooler areas that channels go through downstream. This is thermal wake effect which is characterized in [5]. These aspects (including the thermal wake effect) can be captured by expressing them as thermal resistances: R_{cond} for conduction, R_{conv} for convection and R_{heat} captures fluid flow (as illustrated in figure 3) [2] [6].

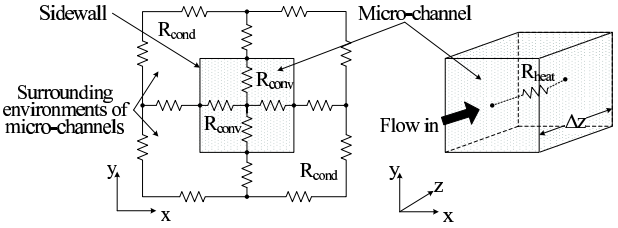


Figure 3: Micro-channel thermal model

Conductive resistance $\mathbf{R_{cond}}$: heat generated in the silicon layer is transferred to the micro-channel sidewalls through conduction, and R_{cond} represents this thermal conduction in the silicon.
Convective resistance $\mathbf{R_{conv}}$: represents the convection from micro-channel sidewalls to the coolant fluid. Assuming the micro-channel has been discretized into grids along the fluid direction z as illustrated in figure 3. The length of each grid is Δz. Let R_{conv} be the convective resistance of one of these sections. It is defined as $R_{conv} = (T^s - T^f)/q$ where q is the heat transferred from sidewalls to the fluid in this section, T^s is the sidewall temperature and T^f is the fluid temperature at this section. As shown in [6] R_{conv} is:

$$R_{conv} = 1/hA \tag{1}$$

where A is the total heat transfer surface area in each section of the channel. Assuming heat can be transferred from all the four sidewalls, the surface area of each section is $A = 2(w_a + w_b)\Delta z$, where w_a and w_b are channel width and height. Here h is convective heat transfer coefficient, which is decided by $h = N_u k_f / D_h$. N_u is the Nusselt number, k_f is the thermal conductivity of fluid and D_h is the hydraulic diameter, which is defined as $D_h = 4 \cdot cross\ sectional\ area/perimeter = 4w_a w_b/(2w_a + 2w_b)$.
Heat resistance $\mathbf{R_{heat}}$: the heat absorbed from sidewalls is then transferred to downstream with the moving flow. The heat transfer with the moving flow is represented by heat resistance R_{heat}:

$$R_{heat} = \frac{1}{C_p \rho f} \tag{2}$$

where f is the volumetric flow rate in each channel and $f = velocity * cross\ sectional\ area = v w_a w_b$ (v is the velocity of fluid in micro-channel). C_p is the fluid specific heat, and ρ is fluid density.

2.2 Micro-channel power consumption

The energy used to flow coolant through micro-channels is a strong function of the level of heat removal desired. Future 3D-ICs are expected to dissipate large amount of power. In order to maintain acceptable thermal levels, increase of chip power dissipation would result in increased power used by the micro-channels, because fluid flow rate needs to be increased to improve the micro-channel cooling performance at the cost of pumping power.

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. The pumping power Q_{pump} is decided by the pressure drop and volumetric flow rate of micro-channels.

$$Q_{pump} = \sum_{n=1}^{N} f_n \Delta P_n \tag{3}$$

Here ΔP_n and f_n are the pressure drop and fluid flow rate of the n-th micro-channel. N is the total number of channels. In this paper, we use single-phase laminar liquid flow as the working fluid. Pressure drop and fluid flow rate are interdependent and also related to other micro-channel parameters such as length and width, etc. The pressure drop in a straight micro-channel is decided by (for the moment we assume that the channel does not have any bends, pressure drop in channels with bends would be discussed subsequently):

$$\Delta P = \frac{2\gamma \mu L v}{D_h^2} \tag{4}$$

Here L is the length of micro-channel, D_h is hydraulic diameter, v is fluid velocity, μ is the viscosity of fluid and γ is a constant determined by the dimension of micro-channel (given in [6]).

Usually fluid pumps are designed to work such that all the micro-channels experience the same pressure drop ΔP. For a given ΔP that the pump delivers across all the channels, fluid velocity v could be estimated by using equation 4. The volumetric fluid flow rate $f = v w_a w_b$ (described in section 2.1) could be estimated as well. Also, fluid flow rate could be controlled by changing the pressure drop. Higher pressure drop results in higher flow rate and therefore better cooling.

2.3 Modeling Mirco-channels with bends

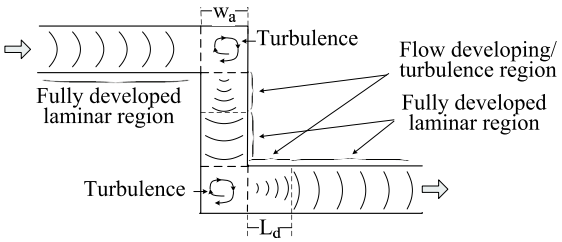


Figure 4: Micro-channel with bends

Consider the channel structure shown in figure 4. The existence of a bend cause a change in the flow properties which impacts 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 4). So a channel with bends has three distinct regions, 1) fully developed laminar flow region 2) the bend corner 3) the developing/turbulent region after the bend [11] [12]. (In a straight channel we only have the fully developed laminar region.) The length of the flow developing region is [13]:

$$L_d = (0.06 + 0.07\beta - 0.04\beta^2) Re D_h \tag{5}$$

where $\beta = w_b/w_a$ is the channel aspect ratio and D_h is the hydraulic diameter explained in section 2.1. Re is the Raynolds number defined by $Re = \rho v D_h/\mu$, where ρ is the mass density of the fluid, v is fluid velocity, and μ is the viscosity of fluid. In this developing region, the cooling effectiveness is analytically impossible to capture and requires large scale numerical simulation to estimate its impact on R_{heat} and R_{conv}. For simplicity, many existing works assume the fluid flow to be fully developed and laminar in this region as well from the point of view of heat transfer, thereby simplifying the analysis without much impact on accuracy [14]. In this paper we also assume that the heat transfer property of fluid in this region is the same as fully developed laminar flow and therefore does not impact R_{heat} and R_{conv}.
Pressure drop: The rectangular bend significantly impacts the pressure drop. Due to the presence of the bend, 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). Let us suppose L is the total channel length, and m is the bend count. Hence $m \cdot L_d$ is the total length that has developing/turbulent flow and $m \cdot w_a$ is the total length attributed to corners (see figure 4). Hence the effective channel length attributed to fully developed laminar flow is $L - m \cdot L_d - m \cdot w_a$. 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 the fully developed laminar region is given in equation 6 [6].

$$\Delta P_f = \frac{2\gamma\mu(L - m \cdot L_d - m \cdot w_a)v}{D_h^2} \quad (6)$$

Here $L - m \cdot L_d - m \cdot w_a$ is the total length of the fully developed laminar region which is explained above, the other parameters are the same as in equation 4. Note that the pressure drop per unit length in the fully laminar region is independent of its location in the micro-channel (distance from the channel inlet).

Pressure drop in flow developing region: The pressure drop in each flow developing region Δp_d is [15]:

$$\Delta p_d = \frac{2\mu v}{D_h^2} \int_0^{L_d} \gamma(z) dz \quad (7)$$

Here $\gamma(z)$ is given by $\frac{3.44}{\sqrt{z/(ReD_h)}}$, where z is the distance from the entrance of developing region in the flow direction. L_d is the length of developing region given by equation 5. Here we assume L_d is less than the size of the grid based on which the micro-channel routing is performed. 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 $\gamma(z)$ and L_d into equation 7 and solving the integration, we can get the total pressure drop of the developing region in this micro-channel:

$$\Delta P_d = m\Delta p_d = mK_d\rho v^2 \quad (8)$$

where $K_d = 13.76(0.06 + 0.07\beta - 0.04\beta^2)^{\frac{1}{2}}$ is a constant associated with the channel aspect ratio β. Please refer to [12] [15] for details.

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

$$\Delta P_{90^\circ} = m\Delta p_{90^\circ} = m\frac{\rho}{2}K_{90}v^2 \quad (9)$$

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 [12].

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:

$$\begin{aligned}\Delta P &= \Delta P_d + \Delta P_f + \Delta P_{90^\circ} \\ &= \frac{2\gamma\mu(L - m \cdot L_d - m \cdot w_a)}{D_h^2}v + m(K_d + \frac{K_{90}}{2})\rho v^2 \end{aligned} \quad (10)$$

As we can see from equations 10, 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 10. Note that we can get two solutions from solving equation 10 in which one is positive valued and the other is negative. The positive solution is the fluid velocity. With the fluid velocity, we can then estimate the volumetric flow rate f, and thus estimate R_{heat} and the pumping power for this micro-channel using equations 2, 3. Hence the pumping power and cooling effectiveness of micro-channels with bends is a strong function of 1) number of bends, 2) location of the channels 3) pressure drop subjected by the pump. Note that when $m = 0$, equation 10 becomes same as equation 4.

Comparing equations 4 and 10, due to the presence of bends, if the same pressure drop is applied on a straight micro-channel and a bended micro-channel of the same length, the bended micro-channel will have lower fluid velocity, which leads to a lower cooling capability due to the increase of R_{heat}. Therefore, to provide sufficient 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.

2.4 Overall thermal model

The thermal behavior of a 3D chip can be modeled by a distributed RC network by partitioning it into fine grids. In this network, each grid is represented by a node. Voltage at each node represents the temperature at that grid. The current source in each grid represents the power dissipated at that chip 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 of each grid to store heat [16]. In this paper we are mostly interested in the steady state thermal behavior, so we focus primarily on resistive networks that represent the thermal behavior of chips. Figure 5(a) illustrates such a resistive network. The values of these resistances can be estimated using well known approaches [16]. For a given resistive network and the power profile, the temperature at each grid can be estimated by solving a system of linear equations (equation 11):

$$GT = Q \quad (11)$$

where G is a matrix decided by the thermal resistance network, Q is a matrix representing the chip power profile and T represents the temperature profile.

As indicated earlier, the thermal behavior of micro-channels can also be modeled by a thermal resistance network (figure 3). The parameters of this resistive network could be computed using the equations described above. Other approaches such as [5] can also be used to generate micro-channel thermal resistive network. Since we assume the cooling effectiveness in bends and developing regions are the same as fully developed laminar region, we can use similar thermal resistive model as straight micro-channels (described in figure 3) for bend and developing regions, in which heat convection from sidewalls to fluid in each grid is modeled by R_{conv} and heat carried by the moving flow is modeled by R_{heat}.

The 3D-IC resistive network and micro-channel network can be easily combined as illustrated in figure 5(b) to generate a unified model that captures the steady state thermal behavior of 3D-ICs with liquid cooling. Other aspects of 3D-ICs such as TSVs can also be incorporated in this resistive network.

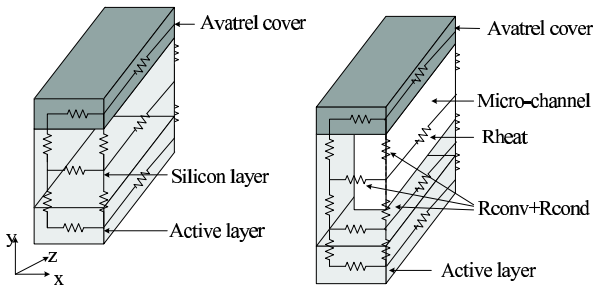


Figure 5: (a)Thermal resistance network of 3D chip, (b) thermal resistance network of 3D chip with micro-channel

3 Micro-channel infrastructure design: Algorithms

Design of cooling infrastructures of engineered systems (building, cars etc) is a very complex process. This is partly because a certain cooling infrastructure choice impacts the cooling performance in a highly nonlinear fashion. Current methods use highly skilled thermal design engineers along with sophistical numerical simulation and optimization tools. Designing 3D-IC micro-channel infrastructure is similarly 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.

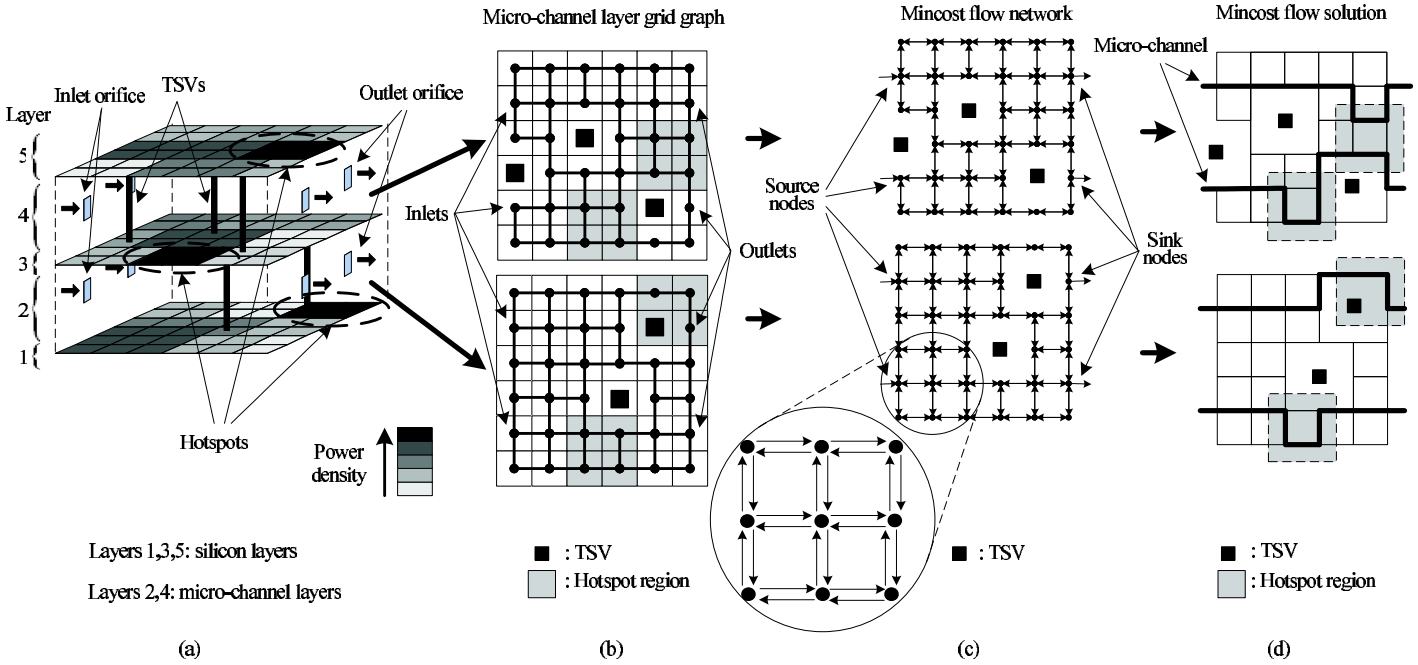


Figure 6: Example of micro-channel infrastructure design using minimum cost flow

$$
\begin{aligned}
& min \quad Q_{pump}(g^l_{i,j}, \Delta P) \\
s.t. & \sum_{\forall j \in I(i)} g^l_{i,j} = 1, \forall \text{grid } i \in \{\text{CI, CO}\}, \forall \text{channel layer } l \\
& \sum_{\forall j \in I(i)} g^l_{i,j} = k \in \{0, 2\}, \forall \text{grid } i \notin \{\text{CI, CO, TSV}\}, \forall \text{channel layer } l \\
& g^l_{i,j} = 0, \text{if grid } i \text{ or } j \in \{\text{TSV}\}, \forall \text{channel layer } l \\
& T^l_i(g^l_{i,j}, \Delta P) \leq T_{max}, \forall \text{grid } i, \forall \text{channel layer } l \\
& g^l_{i,j} \in \{0, 1\}, \forall \text{grids } i, j, \forall \text{channel layer } l \\
& g^l_{i,j} = g^l_{j,i}, \forall \text{grids } i, j, \forall \text{channel layer } l
\end{aligned}
\tag{12}
$$

Figure 6 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 6(a), two intermediate layers comprise of micro-channels). The location 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 6(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 (or a set of grids), then its corresponding neighborhood edges are removed since micro-channels cannot be routed through TSVs (see figure 6(b) for details). Let $g^l_{i,j} = 1$ represents the fact that there is a channel connecting grid i to grid 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 $g^l_{i,j} = g^l_{j,i}$). 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, $I(i)$ 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 either a channel goes through a grid (and therefore $\sum_{\forall j \in I(i)} g^l_{i,j} = 2$) or it does not (and therefore $\sum_{\forall j \in I(i)} g^l_{i,j} = 0$). 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.

3.1 Overall micro-channel infrastructure design flow

This is a very complex problem since 1) the variables need to be discrete 2) the thermal and pumping power models are highly nonlinear. In order to tractably solve this problem we need to develop a systematic methodology. In this paper we investigate such a methodology as illustrated in figure 7. 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. We believe the micro-channel design would be performed after the 3D-IC design process has completed. Hence existing tools for 3D thermal analysis could be used [16].

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.

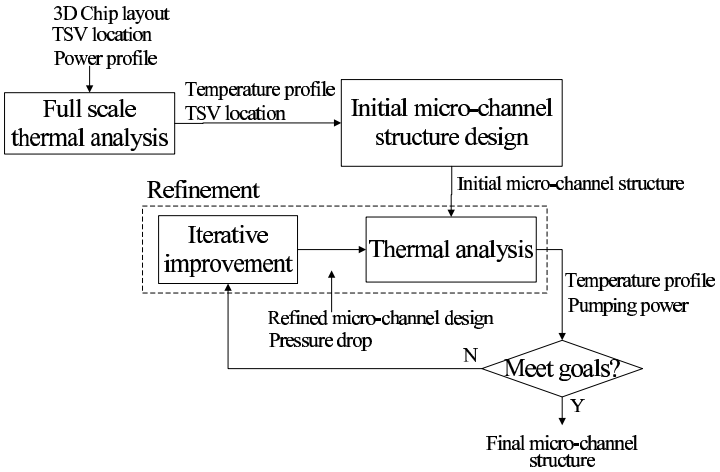


Figure 7: Micro-channel infrastructure design flow

3.2 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 12 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.

Initialization of the minimum cost flow network: Consider the 3D-IC and the corresponding grid graph of each micro-channel layer as illustrated in figure 6(a)(b). For each micro-channel layer, we instantiate a minimum cost flow problem as follows (see figure 6(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 6(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 [17]. The algorithmic details have been omitted for brevity. It is noteworthy that because there is an edge between each pair of neighboring nodes, the flow path could take several bends if necessary.

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 6(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 6(a) could be cooled by two micro-channel layers unlike the top and bottom silicon layers.

As illustrated in figure 6(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 layer 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 from the micro-channel grid just below. If u is set too low then most of the cooling will be done by the channel layer below and vice versa for large u. The u factor must be judiciously chosen since it decides the level of cooling required from a certain micro-channel layer.

In our work, 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 layer, then $u_i^l = 0$ so that all the cooling demand goes to the micro-channel layer right below l (which is in layer $l-1$.

Case 2: If l is the bottommost layer, then $u_i^l = 1$ so that all the cooling demand goes to the micro-channel layer right above l.

Case 3: 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. One might think that uniformly distributing the cooling demand is the optimal choice but the presence of TSVs counters that conventional wisdom. One micro-channel layer may have more TSVs and hence have more constraints towards channel routing. Hence such a micro-channel layer should be assigned less cooling demand. Specifically, 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 8 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 (that is, $T_i^{l-1} > T_{max}$ and $T_i^{l+1} > T_{max}$), the micro-channel layer should provide cooling to both sides (above and below), so the cost is:

$$cost(i,l) = -[(1-u_i^{l+1})T_i^{l+1} + u_i^{l-1}T_i^{l-1}] \tag{13}$$

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 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 8 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

$$cost(i,l) = \begin{cases} -(1-u_i^{l+1})T_i^{l+1}, \text{ if } T_i^{l+1} \geq T_{max} \\ -u_i^{l-1}T_i^{l-1}, \text{ if } T_i^{l-1} \geq T_{max} \end{cases} \tag{14}$$

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.

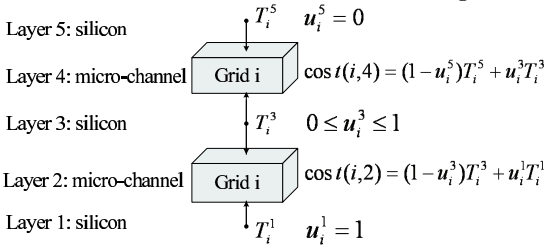


Figure 8: Cost assignment

3.3 Micro-channel refinement

The primary objective of the minimum cost flow formulation is to come up with an initial 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 causes a degradation in the overall cooling quality. In this section we present approaches for iteratively improving the design for improved cooling effectiveness. The micro-channel infrastructure refinement process works as illustrated in figure 7.

3.3.1 Temperature and Pumping Power Analysis

After developing the initial micro-channel infrastructure using the minimum cost flow formulation, we need to do a thermal analysis. The impact of micro-channels on the thermal profile of 3D-ICs 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 channel needs to work at. These parameters are required for estimating the overall 3D-IC thermal profile and also the pumping energy (see discussion in section 2).

As indicated earlier, we assume that all channels are subjected to the same pressure drop by the pump. For a given pressure drop across the pump and the given micro-channel design, equation 10 could be used to determine the velocity and therefore fluid flow rate in each channel. Note that because each channel has different number of bends and different total length, the flow rate would be different too. Based on this flow rate information which is computed for a given pressure drop, the associated R_{heat} could be computed. This information could then be used to estimate the thermal profile of the 3D-IC for a given pumping pressure drop.

We need to estimate the *smallest* pressure drop ΔP (and hence the fluid flow rates) that we need to work at such that thermal constraints of the 3D-IC are satisfied. The smallest pressure drop value ΔP corresponds to the smallest pumping energy. Since all channels are subjected to the same ΔP, this value can be determined by linearly increasing ΔP and calculate the thermal profile for each value until the thermal constraints are met. After finding the minimum required ΔP, we could calculate the pumping power using equation 3. This technique is highlighted in Algorithm 1.

Algorithm 1 Finding the minimum required pumping power

1. $\Delta P = \Delta P_{min}$
2. Repeat steps 3-8:
3. Calculate the fluid velocity using equations 10;
4. Calculate heat resistance R_{heat} using equation 2;
5. Estimate temperature profile using the thermal resistive model described in section **??**;
6. If $\exists i, l, s.t.\ T_i^l > T_{max}$ (T_i^l is the temperature at grid i in layer l);
7. $\Delta P = \Delta P + \delta P$;
8. Else break;
9. Calculate pumping power using equation 3.

3.3.2 Iterative micro-channel optimization

If the pumping power and temperature are unsatisfactory, then the design needs to be optimized. The objective of minimum cost flow formulation did not capture cooling energy and/or number of bends in the channels. Figure 9 illustrates typical situations that can occur. The two micro-channels have significantly different number of bends and cooling demands. For example in figure 9(a) both channels have the same number of bends. Hence for a given pressure drop they will have the same flow rate (see equation 10). But because the second micro-channel needs to remove more total heat, it needs to have a higher flow rate (thereby reducing R_{heat}). Hence we need to increase the pressure drop thereby increase the pumping energy. This action results in increased flow rate in both micro-channels and therefore more than necessary cooling performed by the first micro-channel which is wasteful from an energy perspective. Another situation is illustrated in figure 9(b). One of the channels has significantly higher number of bends and total length (although similar heat load). The higher number of bends and longer length results in smaller flow rate for the same pressure drop and thus in-ability to keep the hotspots cool. Hence the pressure drop needs to increase thereby increasing the pumping energy. The basic idea is that all the micro-channels should have similar levels of heat load, length and number of bends. Hence if a channel has too many bends and 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 un-necessary 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 2 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 measure by the total heat absorbed by the micro-channel, which could be calculated using $q = (T_{out} - T_{in})/R_{heat}$. Here T_{in} is the fluid supply temperature at micro-channel inlet, and T_{out} is the fluid temperature at micro-channel outlet. The R_{heat} here is the heat resistance of that specific channel. Given the pressure drop, power dissipation profile of the 3D-IC and the location and dimensions of the micro-channels, these parameters could be easily calculated (see discussion in section 2). 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 channel i and k.

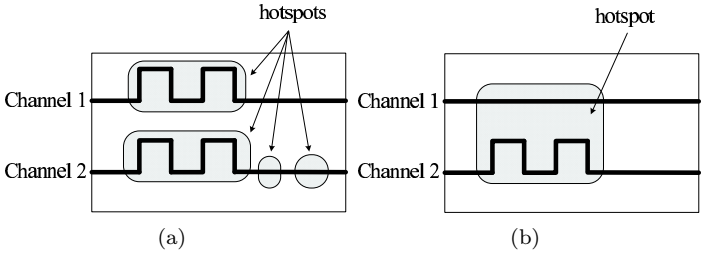


Figure 9: Examples of (a) unbalanced cooling demand, (b) different number of bends

Algorithm 2 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 1;
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 10 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 10). 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 1/2 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 10). 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.

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 1.

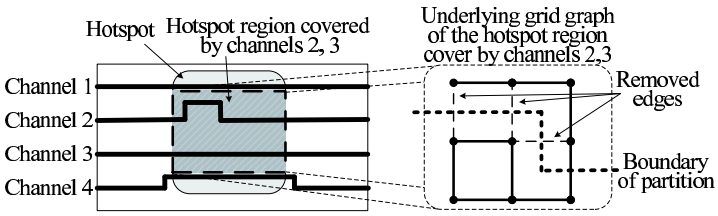


Figure 10: Example of pairwise cooling workload balance

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 2.2, 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 un-necessary 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 11 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 for brevity.

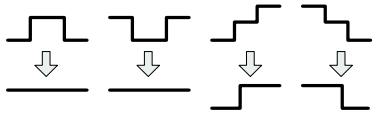


Figure 11: Examples of bend elimination

3.4 Complexity Analysis

Complexity of our algorithm: In our algorithm, the minimum cost flow is easily solvable in polynomial time and many network flow tools can be used to find the optimal solution for minimum cost flow quickly. In the refinement step, the pairwise cooling load balance is basically solving minimum cost flow on a renewed graph in each iteration, and the number of iterations is in proportional to the number of channels. The bend elimination is a pattern match process and its complexity depends on the number of patterns we generate in the library. In our work, we use 22 patterns.

The most time consuming part of this algorithm is thermal analysis. We need to perform thermal analysis at the beginning of our algorithm and also in each iteration of the refinement process. The complexity comes from the fact that we have to re-setup the resistive network every time we change the underlying micro-channel structure or pressure drop ΔP, and then solve equation 11. Although solving equation 11 just involves solving a system of linear equations, the complexity of this process can be high due to the granularity of the grids. Fine grained grids results in more precision at the cost of higher runtime.

Comparison with routing: One might argue that we could use routing for micro-channel infrastructure design. However, the two problems are distinct. Routing basically tries to connect between a set of sources and different areas on chip. But here we would like to distribute the flow so that it **covers** the desired regions in an efficient way so as to provide cooling in different areas. So these two problems are distinct.

In this work, we assume the properties of coolant such as fluid density ρ, viscosity μ, specific heat C_p and thermal conductivity k_f are thermal independent as most of the existing works do [2] [6] [14].

4 Experimental results

We test our method on a two-tier stacked 3D-IC, and each tier contains a micro-channel layer. For the sake of experiment, each tier contains a 4 core CPU. We assume a typical floor plan for each tier. To obtain the power data for each core, we simulated a high performance out-of-order processor with SPEC 2000 CPU benchmarks [18] (using Wattch [19]). For each benchmark, we simulated a representative 250M instructions and sampled the chip power dissipation values using uniform time intervals. We simulated 20 such benchmarks and the resultant power data gave us the power profile for each core of the CPU on each tier. To generate the power profile of a 4 core CPU, we randomly choose 4 of these profiles and arrange them according to the typical floor plan, so each of the resultant power profile represents the power profile on one tier. In the experiment on this 2-tier 3D chip, we choose 2 of these power profiles and each of them represents the power profile on one tier. That is, the benchmark we use in this experiment is a combination of two of these power profiles, each power profile corresponds to one tier.

The area of each chip stack is $1.2 \times 1.2cm^2$, and the grid size is $200 \times 200\mu m^2$ (so 60×60 grids in each layer). The channel dimensions are $w_a = 100\mu m$, $w_b = 400\mu m$. To carry out the thermal analysis, R_{conv} and R_{heat} are estimated using the equations in section 2. We then setup the resistive network by using the hotspot like model in three dimension [16], and solve the system of linear equations in equation 11. The maximum temperature constraint T_{max} is 85°C. The maximum available pressure drop is $500kPa$.

4.1 Comparison with straight channel

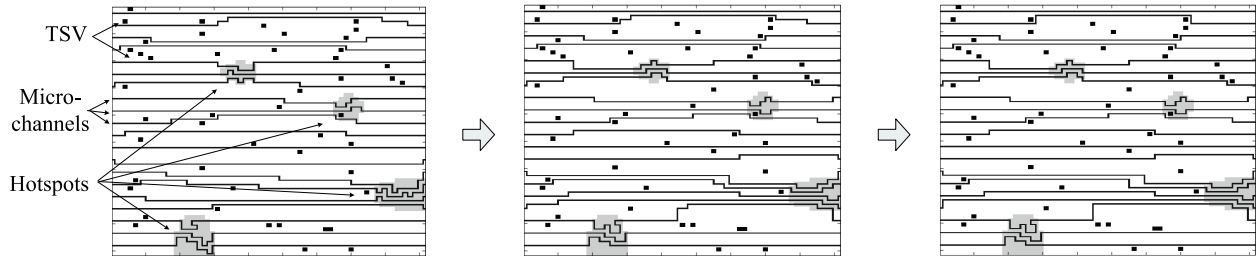
We evaluate our method by comparing our micro-channel design with the micro-channel structure with straight channels. In our design, we use 20 micro-channels. For each benchmark, we tested on different number of TSVs and these TSVs are randomly distributed across each layer of the chip. We find the minimum pressure drop required to cool the 3D chip below thermal constraint for our design and then calculate the associated pumping power Q_{pump}. While in the straight channel design we place as many channels as possible to maximize its cooling efficiency. Note that in the micro-channel system with only straight channels, the number and location of channels are constrained by the TSVs. We also find the minimum required pressure drop and associated pumping power for the straight channel design. The comparison is shown in table 1. Note that for the case when the cooling demand is so large that even using the maximum pressure drop will lead to temperature violation, we label "violation" in these cases.

In the table, Q_{chip} is the total power dissipation for each benchmark, and Q_{pump} is the pumping power, T_{peak} is the maximum temperature achieved under the given pumping power. In the results of our design, we show the results of both the initial design (after solving minimum cost flow), and also the design after pairwise balancing and bend elimination refinement. Here *ratio* denotes the ratio between the pumping power used by our design and that used by the straight channel structure.

As we can see from the table, our algorithm could achieve higher cooling effectiveness compared with the straight channel (with an pumping power saving from 4% to 87%). This is because, the presence of TSV constrained the count and locations of straight channels. Especially when the number of TSVs increases, the available locations for straight micro-channels reduce dramatically. Therefore, to provide sufficient cooling to the hotspots blocked by TSVs, the flow rate should increase significantly. Note that, even though the percentage of grids containing TSVs are small (no more than around 2.1%), its impacts on straight micro-channel design can be significant since the whole row will become unavailable even when there is only one TSV in this row. However, the presence of TSVs will have much less impact on our design. So we could use a smaller pressure drop (flow rates) to provide sufficient cooling.

Table 1: Comparison of our design with straight channel structure (temperature: °C, power: W)

Percentage of grids containing TSV	Benchmark	P_{chip}	Straight channel		Our design						ratio
					Initial design		With pairwise balance		With bend elimination		
			T_{peak}	Q_{pump}	T_{peak}	Q_{pump}	T_{peak}	Q_{pump}	T_{peak}	Q_{pump}	
0.3%	1	149.86	84.86	6.02	84.75	5.25	84.78	4.83	84.62	4.52	0.75
	2	222.76	84.82	15.16	84.69	13.65	84.66	13.42	84.70	12.65	0.83
	3	298.88	84.92	24.16	84.77	23.85	84.75	22.28	84.79	21.08	0.87
	4	372.67	84.88	31.56	84.97	31.25	85.00	31.74	84.99	30.22	0.96
0.7%	1	149.86	84.62	10.40	84.72	5.26	84.89	4.92	84.62	4.65	0.45
	2	222.76	84.44	66.76	84.90	14.08	84.85	13.69	84.96	12.92	0.19
	3	298.88	violation	NA	84.97	24.11	84.95	23.68	84.91	22.59	NA
	4	372.67	violation	NA	84.94	32.95	84.92	32.02	84.97	31.35	NA
1.4%	1	149.86	84.81	43.96	84.74	6.55	84.98	6.23	84.98	5.94	0.13
	2	222.76	violation	NA	84.85	14.28	84.92	13.89	84.96	13.40	NA
	3	298.88	violation	NA	84.92	39.23	84.94	38.78	84.96	38.11	NA
	4	372.67	violation	NA	violation	NA	violation	NA	violation	NA	NA
2.1%	1	149.86	violation	NA	84.96	7.12	84.94	6.98	84.93	6.65	NA
	2	222.76	violation	NA	84.95	18.89	84.98	18.52	85.00	18.10	NA
	3	298.88	violation	NA	84.88	46.64	84.86	46.44	84.89	46.03	NA
	4	372.67	violation	NA	violation	NA	violation	NA	violation	NA	NA



(a) (b) (c)

Figure 12: Resulting micro-channel structure, (a) initial design, (b) with pairwise balancing, (c) with bend elimination

Moreover, as the number of TSVs increases, the improvement of our design over straight channel becomes more significant.

Figure 12 shows the micro-channel infrastructure for benchmark 1 generated by our algorithm. TVSs, which are represented as black squares in the figure, are placed in 1.4% of the grids, . The gray area are hotspot regions. Figure 12(a) shows the initial design generated by minimum cost flow, and figures 12(b) and 12(c) are the refined design after pairwise balancing and bend elimination.

4.2 Impact of the number of TSVs

Figure 13 plots the minimum required pumping power Q_{pump} in our design for benchmark 1 versus the percentage of grids containing TSVs. When the number of grids containing TSVs increases, the pumping power required by our method also increases. This is because the presence of TSVs limits the potential **grids** where micro-channel sections could be places. Especially when there are TSVs in hotspot regions, micro-channels cannot pass through those grids containing TSVs. Therefore, the pressure drop should increase so that micro-channels in neighboring regions can provide cooling to these grids containing TSVs. (Note that in the micro-channel design with straight channels, the presence of TSVs limits the potential **row** where micro-channels could be placed.)

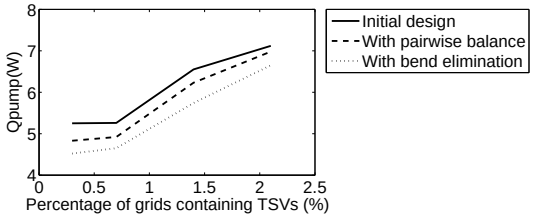


Figure 13: Pumping power versus TSV count

5 Conclusion

In this paper, we investigated the methodology of designing TSV-constrained micro-channel infrastructure. We decide the locations and geometry of micro-channels with bends so that sufficient cooling could be provided using minimum pumping power. Our design could achieve up to 87% pumping power saving compared with the micro-channel structure using straight channels.

References

[1] M. S. Bakir, C. King, and et al, "3D heterogeneous integrated systems: Liquid cooling, power delivery, and implementation," in *IEEE Custom Intergrated Circuits Conference*, pp. 663–670, 2008.

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

[3] J.-M. Koo, S. Im, L. Jiang, and K. E. Goodson, "Integrated microchannel cooling for three-dimensional electronic circuit architectures," *ASME Trans. Journel of Heat Transfer*, pp. 49–58, 2005.

[4] Y. J. Kim, Y. 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.

[5] 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*, pp. 256–263, 2009.

[6] R. W. Knight, D. J. Hall, and et al, "Heat sink optimization with application to microchannels," *IEEE Trans. on Components, Hybrids, and Manufacturing Technology*, pp. 832–842, 1992.

[7] A. K. Coskun, J. L. Ayala, D. Atienzaz, and T. S. Rosing, "Modeling and dynamic management of 3D multicore systems with liquid cooling," in *17th Annual IFIP/IEEE International Conference on Very Large Scale Integration*, pp. 60–65, 2009.

[8] M. B. Healy and S. K. Lim, "Power delivery system architecture for many-tier 3d systems," in *Electronic Components and Technology Conference*, pp. 1682–1688, 2010.

[9] M. Pathak, Y.-J. Lee, T. Moon, and S. K. 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)*, pp. 387–394, 2010.

[10] X.-Q. Wanga, A. S. Mujumdar, and C. Yap, "Thermal characteristics of tree-shaped microchannel nets for cooling of a rectangular heat sink," *International Journal of Thermal Sciences, Vol. 45*, pp. 1103–1112, 2006.

[11] S. 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*, pp. 178–191, 2004.

[12] S. Kandlikar, S. Garimella, and et al, "Heat transfer and fluid flow in minichannels and microchannels," *Elsevier*, 2005.

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

[14] L. Ghodoossi, "Thermal and hydrodynamic analysis of a fractal microchannel network," *Energy Conversion and Management, Elsevier*, pp. 771–788, 2005.

[15] 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.

[16] K. Skadron, M. R. Stan, K. Sankaranarayanan, W. Huang, S. Velusamy, and D. Tarjan, "Temperature-aware microarchitecture: Modeling and implementation," *ACM Trans. on Architecture and Code Optimization*, vol. 1, pp. 94–125, 3.

[17] K. A. Ravindra, L. M. Thomas, and J. B. Orlin, "Network flows: Theory, algorithms and applications," *Prentice Hall*, 1993.

[18] G. Hamerly, E. Perelman, J. Lau, and B. Calder, "Simpoint 3.0: Faster and more flexible program analysis," in *Journal of Instruction Level Parallelism*, 2005.

[19] D. Brooks, V. Tiwari, and M. Martonosi, "Wattch: A framework for architectural-level power analysis and optimizations," in *27th International Symposium on Computer Architecture*, pp. 83–94, 2000.