An efficient channel clustering and flow rate allocation algorithm for non-uniform microfluidic cooling of 3D integrated circuits

Qian, Hanhua; Chang, Chip Hong; Yu, Hao

2013

Qian, H., Chang, C. H., & Yu, H. (2013). An efficient channel clustering and flow rate allocation algorithm for non-uniform microfluidic cooling of 3D integrated circuits. *Integration, the VLSI Journal*, 46(1), 57-68.

<https://hdl.handle.net/10356/94440>

<https://doi.org/10.1016/j.vlsi.2011.12.005>

© 2011 Elsevier B.V. This is the author created version of a work that has been peer reviewed and accepted for publication by *Integration, the VLSI journal*, Elsevier B.V.. It incorporates referee's comments but changes resulting from the publishing process, such as copyediting, structural formatting, may not be reflected in this document. The published version is available at [DOI: <http://dx.doi.org/10.1016/j.vlsi.2011.12.005>].

Downloaded on 26 Apr 2023 17:26:16 SGT

An efficient channel clustering and flow rate allocation algorithm for non-uniform microfluidic cooling of 3D integrated circuits

Hanhua Qian, Chip-Hong Chang, Hao Yu*

*Division of Circuits and Systems, School of Electrical and Electronic Engineering,
Nanyang Technological University, Singapore 639669, Singapore*

**Corresponding author. Tel.: +65 67905873.*

E-mail addresses: qian0028@e.ntu.edu.sg (H. Qian), echchang@ntu.edu.sg (C.-H. Chang), haoyu@ntu.edu.sg (H. Yu).

Abstract

Heat removal problem has been a bane of three dimensional integrated circuits (3DICs). Comparing with other passive cooling techniques, microfluidic cooling appears to be an ideal cooling solution due to its high thermal conductivity and scalability. Without regarding to the fact of non-uniform power distribution of integrated circuits, existing microfluidic cooling with uniform cooling effort incurs large thermal gradient and wastes pump power. This can be avoided by the customized non-uniform cooling scheme proposed in this paper. The microfluidic channels are divided into clusters of relatively homogeneous power distribution and an appropriate flow rate setting is applied to each cluster based on the total flow rate and the maximum allowable temperature of the 3DIC. This paper proposes an efficient clustering algorithm to guide the division of microchannels into clusters and the allocation of cooling resources to each cluster in order to achieve an effective microfluidic cooling with minimal total flow rate. A compact steady state thermal simulator has been developed and verified. Supported by this fast and accurate thermal model, the proposed cooling method and clustering algorithm have been applied to a 3D multi-core testbench for simulation. Compared to the uniform flow rate cooling, the maximum temperature and thermal gradient were reduced under the same total flow rate settings. On the other hand, for a specific peak temperature constraint, up to 21.8% saving in total flow rate with moderate thermal gradients is achieved by the proposed clustered microfluidic cooling.

Keywords: 3DIC; Microfluidic cooling; Clustering algorithm; Channel cluster; Clustered cooling effort; Thermal model

1. Introduction

Three Dimensional Integrated Circuit (3DIC) has gained tremendous research interests recently due to its merits of short interconnection, compact integration and amenability to heterogeneous process technologies [1–7]. Different from the conventional single-layer approach, 3D circuits and systems expand the design space into the third dimension by

stacking multiple layers of integrated circuits. Different layers of active devices can be connected using various methods. One of the most promising techniques is by Through Silicon Vias (TSVs) [8, 9]. Others such as capacitive-coupling-based interconnects [10, 11] are also found in the literature.

Due to physical constraints and product yield requirements, the trend of shrinking device feature size and enlarging die size to integrate more functionality into a single chip has slowed down recently but the consumers' demand for more powerful and ever intelligent computing devices remains. TSV-based 3D integration is most apt to meet this challenge as it makes use of the existing economy-of-scale semiconductor manufacturing processes in an innovative manner. While the topology of 3DIC provides broader design space exploration to overcome many design constraints, it exacerbates the heat dissipation problem. Already heat removal from circuit components to ambient has been a problem for 2D large scale integrations as technology node scales and clock frequency increases [12–16], the stacking of circuits in 3DIC simultaneously increases power density and junction-to-ambient thermal resistances, making the circuits even more vulnerable to thermal induced hazards. Effective and efficient cooling techniques are thus needed to manage the thermal condition of 3D chips.

Traditional heat-sink-on-top cooling method has been proven unsatisfactory to maintain 3D stack's temperature due to its low thermal conductivity and long heat transfer paths for layers far from the heat-sink [17]. To overcome the second drawback, thermal TSVs and thermal wires [18–21] were proposed to provide a fast heat conduit from devices to heat-sink. However, the improvements were shown to be fairly limited and the additional TSVs are overheads that compete for the precious silicon areas. Microfluidic cooling [22] is found to be a more effective cooling technique for 3DIC. By circulating fluidic coolant through inter-layer microchannels, this technique provides local thermal grounds directly to the heat sources, hence eliminates the long heat transfer path problem. In addition, liquid cooling exhibits a much higher thermal conductivity than air cooling by heat-sink, and is scalable to cope with varying number of device layers [17]. All the above advantages have been well documented in the literature [22, 17, 23, 25–27]. One feature of liquid cooling that has escaped the attention is that the flow rates can in principle be controlled individually to provide a customized cooling since they are independent of each other. This is a desired feature for 3D circuits due to the heterogeneity. The power density is usually not evenly distributed either laterally or vertically. Previous works [23, 24, 28–30] assume uniform flow rate distribution among microchannels. With one flow rate for all channels, large flow rate is normally set to cater for the hottest spot to maintain the stack temperature under threshold, and excessive pump energy has been wasted to overcool other parts of the circuit. Although the flow rate can be independently set by allocating a micropump to each channel, doing so will incur large overheads in hardware and control mechanism. A sensible trade-off is to divide the channels into groups and control the flow rate of each group. We call such groupings “channel clusters” by analogy between the control of temperature by microfluidic flow rate and that of power dissipation by voltage scaling in clustered voltage scaling [32]. In [33], we presented a thermal management scheme based on clustered microfluidic cooling. Although an impressive saving of total flow rate with

smoother temperature distribution over the uniform-flow-rate cooling was reported, the channels were grouped intuitively and the results could actually be undermined by poorly grouped clusters.

In this paper, a detailed analysis on microfluidic cooling for 3DICs by different clustered cooling efforts will be presented. A clustering algorithm is proposed to guide the grouping of micro-channels based on the power density distribution. A compact thermal model has also been developed to simulate the steady state temperature of microfluidic cooled 3DICs under different cooling settings. Experiment results show that with proper clustered microfluidic cooling, lower and smoother temperature distribution can be achieved under the same total flow rate constraint. On the other hand, for a given maximum temperature constraint, clustered microfluidic cooling can save up to 21.8% of the total flow rate in our experiment.

The remaining part of this paper is organized as follows. In Section 2, we briefly describe the targeted architecture of 3D integrated circuits. A compact thermal model of microfluidic cooled 3DIC is presented in Section 3. In Section 4, our proposed clustering algorithm to divide the channel clusters based on the power information is introduced. Guideline for achieving minimum total flow rate under maximum temperature constraint is presented. The experiment results are presented in Section 5 before the paper is concluded in Section 6.

2. Architecture of 3D integrated circuit

Fig. 1 illustrates a schematic cross-sectional view of a typical 3D chip that was considered in most publications [23, 28, 29]. In the figure, four active layers are stacked and inter-layer micro-channels are pre-etched before bonding. The active layers are usually composed of Back End Of Line (BEOL), device and thinned bulk layers. The microchannels are assumed to be uniformly distributed and have identical sizes. The channel walls are also equally sized except for the walls at the boundaries. For the ease of modeling, the boundary walls are set to be half the width of the inner walls. The liquid coolants will flow through the micro-channels in one direction as the carrier of the generated heat. A large reservoir of coolant is assumed so that the coolant at the inlet will always be at the same temperature as the ambient.

A multi-core system using the described 3D architecture is used for the simulation in this paper. Each active layer will accommodate four Alpha-2 microprocessors so that in total sixteen cores are combined in the 3D chip. Five benchmarks namely *bzip*, *eon*, *fma*, *gcc* and *gzip* from SPEC2000 [145] are randomly assigned to the microprocessors to simulate real life workload conditions. More details about how this 3D multi-core system is explored for the simulation of our proposed non-uniform flow rate microfluidic cooling scheme can be found later from the experimental settings of Section 5.

3. Compact thermal modeling of microfluid-cooled 3DIC

To develop and verify our proposed clustering algorithm, a compact thermal model is needed to estimate the steady state temperature of 3DICs. Fortunately, there is no need to reinvent the wheel but overcome the deficiency of adapting the existing thermal modeling of microfluid-cooled 3DIC for our purpose. An early work [23] and its enhanced version [24] predict the thermal profiles by solving energy and momentum conservation equations. Iterations are needed until the convergence is met, hence the simulation will be too slow for large systems. More recent work [28] introduces a new concept called “thermal wake” function to account for the influence of upstream heating on the downstream cells. However, extensive numerical pre-simulations are needed to obtain the coefficients for the function. The latest work called 3D-ICE [29] is a 4-resistor-model-based compact transient thermal simulator. The simulation result was claimed to be accurate, but the granularity of grid cell partitioning is too fine to be efficient for reasonably simple circuits and the problem explodes in size for circuits such as the multi-core 3DIC architecture considered in Section 2. This problem was later addressed in [30] by model reduction using 2-resistor-model. Significant reduction in CPU time was reported. However, this model adopts conventional measurements of macro-channels for fluid heat transfer characterization. A survey done by Sobhan and Garimella [31] on microchannel thermodynamics shows different thermal behaviors of microfluids compared with fluids in macro-scale. Thus, 3D-ICE’s approach may introduce deviations from accurate estimation. In [33], the authors have developed a simple thermal model of 3DIC based on HotSpot [46]. However, accuracy is not verified and fluid thermodynamics are taken from measurements of macro-scale setups. This work further improves the previous thermal model and a simple verification is done against the commercial simulator COMSOL [42].

In this section, we briefly introduce our approach to thermal modeling of microfluid-cooled 3DICs. Similar to the existing methods, finite element analysis is used to evaluate the thermal condition of 3DIC. As shown in Fig. 2, the silicon layers are divided into regular grid cells. Every channel is also divided into sections to fit into the grid structure, with the section numbers equal to the row numbers. For each silicon cell at steady state, an equilibrium equation in the form of Eq. (1) can be set up where the first summation indicates the heat taken away by the microchannels and the second summation describes the lateral and vertical heat spreading among the silicon cells:

$$\sum_i (T_{Si} - T_{Ch,i}) g_{Si-Ch,i} + \sum_j (T_{Si} - T_{Si,j}) g_{Si-Si,j} = p_{Si} \quad (1)$$

where T_{Si} denotes the temperature of the silicon cell under investigation. $T_{Ch,i}$ and $T_{Si,j}$ denote the i th channel cell and j th silicon cell in contact with the considered silicon cell. $g_{Si-Ch,i}$ and $g_{Si-Si,j}$ represent the corresponding thermal conductances between the cells. p_{Si} denotes the heat generation rate of the silicon cell, which is the average power consumption. By solving those equilibrium equations simultaneously, the steady state temperature profile can be obtained. Note that although all the solid cells are called “silicon cells” for convenience, they may not necessarily be sole silicon. For instance, for the architecture mentioned in Section 2, the silicon cells are actually composed of BEOL,

device and bulk layers. In such case, composite material properties will be estimated and used.

3.1. Average power consumption estimation

Information on power consumption of 3DIC is a prerequisite for steady state thermal simulation. In this work, the average power consumptions of microprocessors are obtained using Wattch [41]. Wattch is a microarchitecture-level power simulation tool that generates cycle-accurate power values for out-of-order microprocessors according to some specific sets of instructions. In our experiments, SPEC2000 benchmarks are used as the input for the cores. Average unit power consumptions running any benchmark are then estimated by calculating the means of cycle-accurate power values obtained from a complete simulation of the benchmark. For simplicity, the correlation between leakage and temperature is not considered at this stage.

3.2. Thermal conductance network

As indicated in Eq. (1), the key task of building the thermal model is to obtain all the thermal conductances g_{Si-Ch} and g_{Si-Si} . For the case of silicon to silicon cells, the calculation of thermal conductances is straight forward as:

$$g_{Si-Si} = k \cdot A_{Si-Si} / L_{Si-Si} \quad (2)$$

where k represents the thermal conductivity of material, A_{Si-Si} is the contact area between two solid cells and L_{Si-Si} denotes the length of heat transfer path.

Thermal conductances between silicon and channel fluid are influenced by microchannels' dimension and fluid properties such as heat capacity, thermal conductivity, viscosity and flow rate. These factors are summarized by the heat transfer coefficient h , which is formulated as:

$$h = \frac{k_f Nu}{D_H} \quad (3)$$

where k_f is the fluid's thermal conductivity. D_H denotes the hydraulic diameter and is determined by the following equation for rectangular channels:

$$D_H = \frac{4 \cdot A_{Ch}}{P_{Ch}} \quad (4)$$

where A_{Ch} and P_{Ch} are the channel cross-sectional area and channel perimeter, respectively. The Nusselt number Nu is usually obtained by correlations to Reynolds number Re , Prandtl number Pr and channel dimension in the literature. Since our thermal model assumes water as the coolant which has a $Re \ll 2100$ at normal flow rate, laminar flow in the microchannel is most probable [34]. For fully developed laminar flows in microchannels, the following correlation proposed by Peng and Peterson [35] is adopted:

$$Nu_{pp} = 0.1165 \left(\frac{D_H}{Pitch} \right)^{0.81} \left(\frac{H_{Ch}}{W_{Ch}} \right)^{-0.79} Re^{0.62} Pr^{1/3} \quad (5)$$

where $Pitch$ refers to the center-to-center distance of adjacent microchannels. H_{Ch} and W_{Ch} are the height and width of micro-channels. Re and Pr can be obtained by the following formulas:

$$Re = \frac{4 \cdot \rho \cdot Q}{\nu \cdot P_{Ch}} \quad (6)$$

$$Pr = \frac{SpHeat_f \cdot \nu}{k_f} \quad (7)$$

where p , Q , $SpHeat_f$ and ν are the fluid's density, flow rate, specific heat and dynamic viscosity, respectively.

In addition, the heat exchange efficiency is significantly higher at entrance region of microchannels due to thin developing thermal boundary layer [30, 34, 36]. To take the entrance effect into consideration, we adopt the formulation proposed by Seider and Tate [36] as follows:

$$Nu_{ST} = Nu_0(Re \cdot Pr)^{1/3} \left(\frac{D_H}{l} \right)^\gamma \quad (8)$$

where l represents the distance from channel entrance. As Seider and Tate's correlation was derived from macro-scale channels, its original coefficients ($Nu_0=1.86$, $\gamma=1/3$) need to be revised to adapt to micro-scale cases. With the aid of COMSOL [42], the new coefficients are model-fitted to obtain $Nu_0=30$, $\gamma=1.67$, which implies a faster decay of heat transfer coefficient at entrance region for a shorter entrance length.

The aggregated Nusselt number can thus be calculated as:

$$Nu = Nu_{pp} + Nu_{ST} \quad (9)$$

A plot of Nusselt number is shown in Fig. 3. The microfluid exhibits high heat transfer coefficient initially at the entrance region and decays asymptotically to a constant as the thermal boundary profile is fully developed.

The convective conductance g_{Si-Ch} can thus be calculated as:

$$g_{Si-Ch} = h \cdot A_{Si-Ch} \quad (10)$$

where A_{Si-Ch} denotes the effective heating area from the silicon cell to the channel section.

3.3. Steady state thermal analysis

From the above formulas, all the thermal conductances can be easily obtained. Note that thermal conductances between cells will be zero if no direct heat exchange is observed. By rearranging Eq. (1) and assuming that there are N silicon cells and M microchannel sections in total, for a silicon cell i , the following equation holds:

$$\begin{aligned}
& \left(\sum_j g_{Si-Si,i,j} + \sum_k g_{Si-Ch,i,k} \right) T_{Si,i} \\
& \underbrace{-g_{Si-Si,i,0} T_{Si,0} - \cdots - g_{Si-Si,i,N-1} T_{Si,N-1}}_{\text{excluding term with } T_{Si,i}} \\
& -g_{Si-Ch,i,0} T_{Ch,0} - \cdots - g_{Si-Ch,i,M-1} T_{Ch,M-1} = p_i
\end{aligned} \tag{11}$$

Notice that the temperatures of microchannel sections are yet to be determined. In fact, at steady state, the temperatures of every microchannel sections are also held constant and a balance is established between the incoming heat and heat loss for each section. This thermal balance can be formulated as:

$$\sum_k (T_{Si,j,k} - T_{Ch,j}) g_{Si-Ch,j,k} = (T_{Ch,j} - T_{Ch,j-1}) \rho Q \cdot SpHeat_f \tag{12}$$

where the left side of the equation expresses the total amount of heat extracted from the silicon cells, and the expression on the right side represents the cooling effect produced by the coolant from the upstream section. $T_{Si,j,k}$ denotes the temperatures of the silicon cells surrounding the j th section of a channel and $T_{Ch,j-1}$ denotes the temperature of the upstream microchannel section. With the additional M equations for the microchannel sections, the sparse linear system $G \cdot T = P$ can now be solved. The column matrix T stores the temperature values of every cell in the 3DIC.

3.4. Software implementation

Our thermal model is programmed in C language, which invokes KLU sparse solver [43] provided by SuiteSparse [44]. The thermal model takes a configuration file as input for 3DIC structure description. Based on this description, the conductance matrix G is constructed by filling in the column pointer array, row index array and numerical value array required by KLU. The heat generation matrix P is created by padding the power consumption of each grid cell. As the wall and channel cells takes in external coolant at ambient temperature, they will have zero entries in P matrix except for the inlet channel cells which has entries of $Q \cdot SpHeat_f \cdot T_a$, where T_a is the ambient temperature. The linear system will then be solved by KLU to obtain the temperature matrix.

The processes of building conductance matrix and heat generation matrix both have a time complexity of $O(m)$, where m refers to the number of nonzero entries. In addition, the solver of KLU has a time complexity of $O(n+m)$, where n denotes the matrix size. Note that for thermal modeling of 3DIC, m is in the order of $O(n)$. Thus our thermal model has an overall time complexity of $O(n)$ which scales linearly with total number of grid cells.

4. Proposed channel clustering algorithm for non-uniform microfluidic cooling effort

Depending on the circuit function and floorplanning, the power consumption of 3D stack is usually distributed unevenly on each layer. Fig. 4 shows the top layer's power

density map of the target 3D multi-core architecture, where the red colored cells represent high power density areas as opposed to the blue regions where the power densities are relatively low. If uniform cooling effort is applied to the 3D stack, to keep the temperatures of potential hot spots under threshold, a large flow rate needs to be provided. This will result in wasted energy for over-cooling of the low power regions, which creates large lateral thermal gradients and shortens the device's lifespan. In contrast, if the cooling effort can be customized to adapt to the appropriate cooling demand variation caused by the uneven power distribution, efficient microfluidic cooling with better steady state results is envisioned.

The ultimate control of cooling effort, i.e., the microchannel flow rates, requires one flow rate setting per channel, which leads to large overheads and is usually unnecessary. Instead, a limited number of flow rate settings can be assigned to different channels whereby the microchannels that share the same setting are said to belong to the same channel cluster. An algorithm that guides the clustering of channels and appropriate flow rate assignment is hence needed to project an optimal channel allocation for this notion of microfluidic cooling.

4.1. Clustering of channels

The heat exchange paths of any silicon cell in the 3D grid can be classified into two categories, as implied by Eq. (1). One is referred to as the silicon to channel heat transfer which contributes to microfluidic cooling. The other is referred to as the silicon to silicon heat transfer which only helps to spread the heat generation over the entire chip. The former can be controlled directly by varying the flow rate, while the latter depends only on the nearby thermal gradient due to the fixed silicon to silicon thermal conductance for a fixed architecture. However, the silicon to channel heat transfer is predominantly contributed by the larger temperature differences and highly conductive liquid cooling. An even temperature profile, as part of the goal of our proposed cooling method, will further reduce the silicon to silicon heat transfer to result in nearly zero thermal gradient. Therefore, in steady state, the heat generated by each silicon cell needs to be absorbed locally by the neighboring microchannels. In a nutshell, the cooling effort of microchannels should match the power distribution of the silicon cells. Hence, the power consumption level to be coped with by each channel makes a good criterion to divide the channel clusters.

As shown in Fig. 2, any microchannel (denoted by the channel number l_i , $i=0, 1, \dots, M-1$) is in contact with two columns of silicon cells that generate heat, one above and one below the channel. We define the two columns of silicon cells as the cooling region of channel l_i , denoted as $CR(l_i)=\{\text{silicon cell } j / j \text{ is in contact with channel } l_i\}$.

Let the power consumption of silicon cell j be P_{Cj} . It is possible to define a cooling demand for each channel as a weighted sum of power dissipation based on the silicon cells in the region, i.e., $P_{sum,l_i} = \sum_{j \in CR(l_i)} (\alpha_j \cdot P_{Cj})$ where

$$\alpha_j = \begin{cases} 0.5 & \text{if cell } j \text{ is on an inner layer} \\ 1 & \text{if cell } j \text{ is on the top or bottom layer} \end{cases}$$

The factor α accounts for the advantage of cells situated at the inner layers that have twice the number of channels for cooling compared to those at the top or bottom layer. These cooling demands are then sorted in ascending order of magnitude into an array $P = [p_0, p_1, \dots, p_{M-1}]$, where $p_i < p_j$ for $i < j$ and $p_i = P_{sum, l_i}$ for $i, l_i \in [0, M-1]$. The channel indexes corresponding to the sorted power values in P can be retrieved from an order array, $L = \{l_0, l_1, \dots, l_{M-1}\}$, where $l_i = j$ if $p_i = P_{sum, j}$.

The next step would be the segmentation of the sorted array P to divide the M channels into N ($N < M$) clusters. Since channels in one cluster will share the same flow rate setting, the standard deviation of their P_{sum, l_i} values should be as low as possible to achieve an uniform cooling under the same flow rate. This can be readily resolved by finding the natural separations, which are the largest differences between every pairs of adjacent elements of the array P . An array $D = [d_0, d_1, \dots, d_{M-2}]$ can be obtained by computing the difference $d_i = p_{i+1} - p_i$ for $i = 0, 1, \dots, M-2$. The demarcations between clusters can then be found by searching for the $(N-1)$ largest elements in D .

The number of clusters, N can create subtle trade-off between conflicting factors such as the thermal profile, total flow rate and overhead for a given power trace. The dilemma is that there is no easy way to predetermine an ideal N . Therefore, a divide and conquer approach is adopted in this work. By searching for the largest difference in D , the elements in the sorted array P are divided into two clusters. If the largest difference is d_j , then all channels with indexes l_i for $i \in [0, j]$ will be grouped into one cluster and the remaining channels with indexes l_i for $i \in [j+1, M-1]$ will be grouped into another cluster. Let $r_i = p_{M-1} - p_i$ be the moderated cooling demand of the i th channel, and μ_R and σ_R be the mean and standard deviation, respectively, of the modulated power values of all channels within the same cluster. If $\sigma_R > \eta \cdot \mu_R$, where η is a parametric fraction, the cluster will be further divided into two clusters by the same approach. This process iterates until every clusters have their σ_R less than $\eta \cdot \mu_R$. The pseudo-code of the clustering algorithm is shown in Algorithm 1.

Algorithm 1. A divide and conquer algorithm for channel clustering.

Function clustering(L, η).

Input: L : Set of channels to be clustered; η : cluster control fraction.

Output: C : Set of channels in a cluster;

P_{Ci} : Power consumption of cell i ;

$CR(l_i)$: Cells in contact with channel $l_i \forall i \in [0, M-1]$

$M = n(L)$: Number of channels in L ;

for $l_i = 0$ to $M-1$ **step 1** **do**

$$P_{sum, l_i} = \sum_{j \in CR(l_i)} (\alpha P_{c_j});$$

end for

```

Sort  $P_{sum,l_i}$ :  $P=\{p_i\}_{i=0}^{M-1}$ , where  $p_i \leq p_{i-1} \forall i \in [0, M-2]$ ;
Order  $L=\{l_i\}_{i=0}^{M-1}$ :  $l_i \leftarrow j$  if  $p_i=P_{sum,j}$ ;
Calculate  $R=\{p_{M-1}-p_i\}_{i=0}^{M-1}$ ;
for  $i=0$  to  $M-2$  step 1 do
     $d_i = p_{i+1}-p_i$ ;
end for
 $D=\{d_i\}_{i=0}^{M-2}$ ;
 $t=\arg\{\max_j d_j\}$ ;
 $C1=\{l_i\}_{i=0}^t$ ;  $C2=\{l_i\}_{i=t+1}^{M-1}$ ;
Calculate  $\mu_{R1}$  and  $\sigma_{R1} \forall l_i \in C1$ ;
if  $\sigma_{R1} < \eta \cdot \mu_{R1}$  then
    return  $C1$ ;
else
    return clustering( $C1, \eta$ );
end if
Calculate  $\mu_{R2}$  and  $\sigma_{R2} \forall l_i \in C2$ ;
if  $\sigma_{R2} < \eta \cdot \mu_{R2}$  then
    return  $C2$ ;
else
    return clustering( $C2, \eta$ );
end if

```

The reason for using a biased r_i value instead of the p_i value directly for the computation of mean and standard deviation is to relax the uniformity criterion for clusters with lower power consumption while forcing a tighter homogeneity in power consumption for high power regions. This is because high power regions often have low population with high variation in power values. If the clustering is not carried out with enough granularity, severe mismatch between the cooling effort and cooling demand may occur in these regions and lead to potential thermal management failure. The control parameter η provides the desire trade-off between cooling efficiency and overheads. If η is set too high, the clustering may not be optimal; and if η is set too low, excessive clustering may incur heavy overheads. In our simulation, the number of clusters is constrained to between two and six.

4.2. Allocation of flow rates

The power profile of a 3D chip often accords with a Pareto distribution, i.e., there is a large number of low power areas and a few patches of high power points. As a result, the channel clustering algorithm will normally produce some clusters with many microchannels of low cooling demand and a few clusters with much less microchannels that require high cooling effort. If the cooling resource is constrained by the total available flow rate, the basic principle of its reallocation is just to move the flow rate saved from low-cooling-demand clusters to where it is needed the most. From the system point of view, the amount of heat removed is $(T_{out} - T_{in}) \cdot Q \cdot SpHeat_f$. If the flow rate

is ideally adjusted, uniform temperature distribution can be achieved whereby the inlet and outlet temperature differences of micro-channels are similar. In that case, the flow rate should be proportional to the amount of heat removed, which is equal to the steady state power dissipation. Quantitatively, the average cooling demand $\bar{P}(C_i)$ of any cluster C_i can be obtained by calculating the mean of P_{sum,l_j} for all channels $l_j \in C_i$. Due to the predominant silicon to channel heat transfer, the following approximation is used to apportion the total flow rate to each cluster for the best cooling effect:

$$Q(C_0) : Q(C_1) : \dots : Q(C_{N-1}) \approx \bar{P}(C_0) : \bar{P}(C_1) : \dots : \bar{P}(C_{N-1}) \quad (13)$$

Thus, the flow rate per channel for the i th cluster is set to be

$$Q(C_i) = Q_{total} \cdot \frac{\bar{P}(C_i)}{\sum_{j=0}^{N-1} n(C_j) \bar{P}(C_j)} \quad (14)$$

where Q_{total} denotes the total available flow rate and $n(C_j)$ denotes the number of channels in Cluster C_j .

4.3. Minimization of cooling effort under thermal constraint

In most thermal management scheme, a maximum allowable temperature is set as the control target instead of a fixed total flow rate [28, 26]. As clustered microfluidic cooling is capable of producing an evenly distributed temperature profile, the cooling effort can be reduced and the total flow rate can be minimized effectively. In our proposed scheme, this is accomplished by incorporating the thermal modeling metric into the clustering algorithm. The procedure is described as follows:

- *Step 1:* Perform channel clustering according to the power information.
- *Step 2:* Simulate the steady state temperature of the 3D stack.
- *Step 3:* Record the current maximum steady state temperature of silicon cells as T_{max} .
- *Step 4:* Determine scaling factor.

$$\varphi = \frac{\overline{T_{max}(Ch)} - T_a}{\overline{T_{max}(Ch)} + (T_{th} - T_{max}) \cdot \left(1 - \frac{g_{Si,y} + g_{Si,wall}}{nr \cdot g_{Si,Ch}} \right) - T_a} \quad (15)$$

where $\overline{T_{max}(Ch)}$ denotes the average temperature of the channel sections that are in contact with the silicon cells of maximum temperature, and T_{th} denotes the temperature threshold. $g_{Si,y}$, $g_{Si,wall}$ and $g_{Si,Ch}$ denote the thermal conductances from silicon circuit cell to silicon circuit cell along the channel direction, from silicon circuit to channel wall cell, and from silicon circuit cell to channel cell, respectively.

- *Step 5:* Update the flow rate settings to $Q_{new} = \varphi \cdot Q_{old}$.

The scale factor φ provides a rough guide on the minimization of cooling effort. When flow rate is minimized, the maximum circuit temperature will approach the threshold. The temperature of channel cells in contact with the hottest circuit cell will also rise accordingly to remove the same amount of heat at low flow rate. Therefore, the flow rate is inversely proportional to the inlet–outlet temperature difference. In Eq. (15), the temperature rise of channel cells in contact with the hottest spot is estimated as a scaled

temperature rise of the hottest spot. The use of thermal conductance at high flow rate tends to overestimate the channel temperature while the inclusion of silicon circuit to wall flux tends to underestimate the channel temperature. This kind of offset makes the estimation relatively accurate.

As such, the total flow rate is reduced by $(1 - \phi)$. However, since the overall steady state temperature of 3D stack is raised, the thermal gradient will be amplified as well. A positive effect of such amplification is that higher thermal gradient induces a bigger heat flux among the silicon cells to swiftly spread the temperature away from the hot spots. This has added a safety margin spontaneously into our minimization method to ensure that the maximum temperature will not exceed the thermal limit.

4.4. Discussion on practical implementation

Our proposed clustered microfluidic cooling for 3DIC requires coolant supply at multiple flowrates simultaneously. Intuitively, an easy way of implementation is to provide each cluster with an independent pump and manifold system. However, doing so would incur excessive overhead on design complexity, space and power consumption. A more feasible way is to supply the coolant for one 3DIC or several 3D stacks with only one micro-pump, while using valves to achieve different flow rate settings as suggested in [37]. Micropumps such as piezo-electric micropump [39] and miniature valves [40] are widely available in the market. In such case, the overhead is only the increased complexity of manifold system and additional valves for multiple flow rate settings. With a restricted number of clusters, this overhead is justifiable by the longer lifespan, stable operation and possible speedup of 3DIC in the long run.

As the clustering of channels will be implemented in hard-ware, once decided, it will be almost impossible to change the channel groupings during runtime. Therefore, the power profile needs to be chosen such that the resulting clustering addresses most of the applications executed on the 3DIC. A reasonable solution is to consider typical applications executed by the 3DIC. Fortunately, the optimal distribution of microchannels will vary much less than the power variations for different applications. Thus, the minor mismatch between cooling and power dissipation would not be catastrophic during runtime. This problem will be discussed further in Section 5.

5. Experiment results and discussion

In this section, we validate our thermal model against the commercial simulator COMSOL [42] based on a 2-layer 3D stack and evaluate the effectiveness of our proposed clustering algorithm. A 4-layer multi-core architecture consisting of 16 Alpha-2 cores with four cores per layer is selected as a test platform. SPEC2000 applications *bzip*, *eon*, *fma*, *gcc* and *gzip* [45] are randomly assigned to the cores as inputs. Power information of the cores is collected through Wattch [41] simulation. To show the effect of the clustering algorithm, four rounds of random allocation of the benchmarks to cores and one round of average power assignment are performed to generate five different power distributions for the case study. For all the case studies, the steady state power

density varies between 2.98 W/cm^2 and 68.95 W/cm^2 across the entire 3D stack. Detailed sizing of both architectures and corresponding experimental settings are summarized in Table 1. The proposed clustering algorithm and the thermal model are both implemented in C/C++. In the absence of physical prototypes, all the experiments are performed based on simulations on a Linux server with Intel Xeon 3.47 GHz processor and 50 GB of memory

5.1. Verification of thermal model

A smaller $1.8 \text{ mm} \times 1.8 \text{ mm}$ 2-layer stack is used as the test bench for this verification. Uniform power density is assumed and different power density levels are explored with different flow rate settings. As shown in Fig. 5, the top layer thermal profiles from our model and COMSOL are plotted and compared. Due to a uniform heat flux input, the temperature varies only along the channel direction. Both graphs are hence divided into nine rows and the average temperature of each row is compared in Table 2 in the ascending order of temperature. A maximum of 2.34% of average temperature deviation is observed.

The maximum and minimum temperatures in Kelvin obtained from our thermal model and COMSOL are further reported in Table 3. It can be seen that the maximum temperature is accurately predicted with an average error of only 0.46%. However, discrepancies of minimum temperatures are observed for the high power density and low flow rate cases. The error is mainly contributed by the finite and uniform grid division adopted by our thermal model. Note that the minimum temperature always appears at the inlet area where the heat transfer coefficient varies significantly along the flow direction. COMSOL adopts an adaptive mesh construction method so that very fine-grained grid division is performed at critical places such as the inlet area. Thus it is able to capture the very high heat exchange rate right after the inlet. Our model adopts uniform grid division method for compact thermal modeling, which underestimates the heat transfer at the inlet area. As shown in Fig. 6, the prediction of minimum temperature becomes more accurate when finer grid division along the flow direction is adopted, which is actually a trade-off between accuracy and runtime. However, in most cases, the focus is on the hottest spot instead of the coolest point. Thus grid division at average granularity would be sufficient for most thermal modeling tasks.

In addition, our model is fast in computation. At default granularity of grid division, the running time is too short to be recorded by the clock () function. Thus we deliberately increase the problem size to 12 960 by increasing the row division to 288. The runtime at this configuration is about 0.07s, which is $2142 \times$ faster than the 150 s taken by COMSOL to solve the same problem. In a nutshell, our thermal model produces reliable simulation results in short runtime to corroborate the effective-ness of our clustering algorithm.

5.2. Clustering of microchannels

As described in Section 4, the total heat generation to be handled by each microchannel in the five study cases is determined. In Cases 1–4, the five benchmarks are randomly distributed to the 16 cores, while in Case 5, all the cores are assigned with an average power consumption which is the mean value of the five benchmarks. The last case is presented to simulate the practical case where clustering is performed with an average power consumption to cater to various runtime scenarios. The sorted power values in ascending order of magnitude are charted in Fig. 7. To be concise, each dot in the chart actually represents the power level of a channel bundle of five channels sharing the same cooling region.

The clustering results at different η values are summarized in Table 4, where the number of channels in each cluster is listed in ascending order of cooling demand. As the value of η decreases, the deviation in the total power dissipations among channels in the cluster reduces. It can be seen that more clusters are defined with a stringent variation control while coarse clustering is resulted when η is relaxed. To have the best cooling effect while not imposing excessive overhead, η is set to be 20% for Cases 1–4 and 10% for Case 5.

5.3. Uniform cooling effort v.s. clustered cooling effort

With the above clustering results, the fixed total flow rate of 288 ml/min are reallocated to each cluster, as shown in Table 5, where the flow rate per channel allocated to each cluster is listed in ascending order of cooling demand. Here the upper limit for flow rate per channel is specified as 1.5 ml/min corresponding to about 90 KPa pressure drop according to the method of calculation in [38].

Fig. 8 depicts the top layer's steady state thermal maps in Case 1 under both uniform and non-uniform flow rate coolings by channel clustering. It is clear that with clustered microfluidic cooling, the overall temperature of the 3D chip is much lower. Numerical results for all the study cases are summarized in Table 6. Note that for Cases 1–4, the power distribution and channel clustering are ideally matched as the same power profile is used for both clustering and thermal simulation. While for Case 5, power profiles of Cases 1–4 are used for thermal simulation after clustering based on average power consumption. Only the maximum temperature is recorded because the minimum temperature is always close to 300 K at the inlet area. From the table, it is clear that clustered cooling method is effective in reducing the peak temperature under both ideal and practical cases. On average, a drop of 6.22 °C in the maximum temperature is observed.

5.4. Minimization of cooling effort under thermal constraint

Based on the steady state simulation results, the total cooling effort can be further minimized by the proposed minimization steps presented in Section 4.3. By finding the hottest point of the 3D stack under clustered microfluidic cooling and its nearby channel

temperatures, the scaling factors ϕ are calculated and the steady state thermal conditions are re-simulated after reducing every channel's flow rate by $(1-\phi)$. The maximum temperatures after minimization are reported in Table 7. It can be seen that the minimization method accurately predicts the flow rate demand under certain thermal constraint with a suitable safety margin in both ideal and practical cases. For the purpose of comparison, uniform flow rate thermal management is also implemented by a recursive decrease of flow rate until the maximum temperature of the 3D stack approaches the maximum temperature limit of 358.15 K (85 °C). For a fair comparison, the maximum temperature after minimization is further pushed towards the thermal limit by reducing the flow rate proportionally. Some key findings are summarized in Table 8, where the total flow rate is measured in ml/min. It is shown that with clustered cooling scheme, the total flow rate can be saved up to 21.8% under the same thermal constraint.

In addition, Fig. 9 compares the top layer thermal distribution of two minimization methods for Case 1. With the uniform cooling scheme, most of the chip area is overcooled to keep the center hot spot within threshold. Although the thermal map of our proposed method exhibits high temperature in the downstream region, none of the spots exceeds the thermal limit. Furthermore, with uniform flow rate, there is no clue on how to adjust the flow rate to maintain the maximum temperature just below the thermal limit (hence it was accomplished by recursive trial and error). In contrast, our proposed method provides a logical way to scale flow rates according to the maximum allowable temperature. This is useful for runtime thermal management controller.

6. Conclusion

This paper has proposed a novel channel clustering algorithm to guide the grouping of microfluidic channels for a customized fine-grained cooling based on power distribution of 3DICs. A thermal model has also been built and verified in order to facilitate the reallocation of flow rate to optimize the cooling effort based on the maximum allowable temperature. A 3D multi-core architecture is selected as a vehicle to demonstrate the clustering process. From the experiment results, it can be concluded that clustered microfluidic cooling guided by the proposed channel clustering and flow rate allocation algorithms effectively reduces the peak temperature and thermal gradients. In addition, our simulation results show that the proposed cooling effort minimization technique can save up to 21.8% of total flow rate compared to the uniform flow rate cooling without exceeding the prespecified thermal limit of the 3D stacks.

Acknowledgement

This project is partially sponsored by the Singapore Ministry of Education Academic Research Fund Tier-2 Grant ARC 5/11 and Nanyang Technological University Start-up-grant M58040016. The authors would also like to thank Mr. Xiwei Huang for his support on power simulation used in this work.

References

- [1] P. Jacob, et al., Mitigating memory wall effects in high-clock-rate and multicore CMOS 3-D processor memory stacks, *Proceedings of the IEEE* 97 (1) (2009) 108–122.
- [2] L.P. Carloni, P. Pande, Y. Xie, Networks-on-chip in emerging interconnect paradigms: advantages and challenges, in: *Proceedings of the Third ACM/ IEEE International Symposium on Networks-on-Chip*, San Diego, California, 10–13 May 2009, pp. 93–102.
- [3] Y. Ye, et al., 3D optical networks-on-chip (NoC) for multiprocessor systems-on-chip (MPSoC), in: *Proceedings of the IEEE International Conference on 3D System Integration (3DIC)*, San Francisco, California, 28–30 September 2009, pp. 1–6.
- [4] B. Feero, P. Pande, Networks-on-chip in a three-dimensional environment: a performance evaluation, *IEEE Transactions on Computers* 58 (1) (2009) 32–45.
- [5] Y.F. Tsai, Y. Xie, N. Vijaykrishnan, M.J. Irwin, Three-dimensional cache design exploration using 3DCacti, in: *Proceedings of the IEEE International Conference on Computer Design (ICCD)*, San Jose, California, 2–5 October 2005, pp. 519–524.
- [6] G.H. Loh, Y. Xie, B. Black, Processor design in 3D die-stacking technologies, *IEEE Micro* 27 (3) (2007) 31–48.
- [7] H. Hua, et al., Exploring compromises among timing, power and temperature in three-dimensional integrated circuits, in: *Proceedings of Design Automation Conference*, San Francisco, California, 24–28 July 2006, pp. 997–1002.
- [8] J.A. Burns, et al., A wafer-scale 3-D circuit integration technology, *IEEE Transactions on Electron Devices* 53 (10) (2006) 2507–2516.
- [9] A.P. Karmarkar, X. Xu, V. Moroz, Performance and reliability analysis of 3D-integration structures employing Through Silicon Via (TSV), in: *Proceedings of 2009 IEEE International Reliability Physics Symposium*, Montreal, Canada, 26–30 April 2009, pp. 682–687.
- [10] E. Culurciello, A.G. Andreou, Capacitive coupling of data and power for 3D silicon-on-insulator VLSI, in: *Proceedings of 2005 IEEE International Symposium on Circuits and Systems*, vol. 4, Kobe, Japan, 23–26 May 2005, pp. 4142–4145.
- [11] M. Scandiuazzo, et al., 3D system on chip memory interface based on modeled capacitive coupling interconnections, in: *Proceedings of 2010 IEEE International Conference on 3D Systems Integration Conference (3DIC)*, Munich, Germany, 16–18 November 2010, pp. 1–4.

- [12] D. Brooks, M. Martonosi, Dynamic thermal management for high-performance microprocessors, in: IEEE International Symposium on High-Performance Computing Architecture, Monterrey, Nuevo Leon, Mexico, 19–24 January 2001, pp. 171–182.
- [13] W.P. Liao, L. He, K.M. Lepak, Temperature and supply voltage aware performance and power modeling at microarchitecture level, IEEE Transactions on CAD of Integrated Circuits and Systems 24 (7) (2005) 1042–1053.
- [14] J. Donald, M. Martonosi, Techniques for multi-core thermal management: classification and new exploration, in: Proceedings of IEEE International Symposium of Computer Architecture, Boston, Massachusetts, 17–21 June 2006, pp. 78–88.
- [15] A. Kumar, L. Shang, L.-S. Peh, N.K. Jha, System-level dynamic thermal management for high-performance microprocessors, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 27 (1) (2008) 96–108.
- [16] T. Ebi, M. Abdullah, A. Faruque, J. Henkel, TAPE: thermal-aware agent-based power economy for multi/many-core architecture, in: Proceedings of IEEE/ ACM International Conference on Computer-Aided Design, San Jose, California, 2–5 November 2009, pp. 302–309.
- [17] M.S. Bakir, B. Dang, J.D. Meindl, Revolutionary nanosilicon ancillary technologies for ultimate-performance gigascale systems, in: Proceedings of Custom Integrated Circuits Conference, San Jose, California, 16–19 September 2007, pp. 421–428.
- [18] H. Yu, Y. Shi, L. He, T. Karnik, Thermal via allocation for 3D ICs considering temporally and spatially variant thermal power, IEEE Transactions on Very Large Integrated Circuits and Systems 16 (12) (2008) 1609–1619.
- [19] H. Yu, J. Ho, L. He, Allocating power ground vias in 3D ICs for simultaneous power and thermal integrity, ACM Transactions on Design Automation of Electronic Systems 14 (3) (2009) 1–31.
- [20] J. Cong, Y. Zhang, Thermal via planning for 3-D ICs, in: Proceedings of IEEE/ ACM International Symposium Computer-Aided Design, San Jose, California, 6–10 November 2005, pp. 744–751.
- [21] T. Zhang, Y. Zhang, S. Sapatnekar, Temperature-aware routing in 3-D ICs, in: Proceedings of IEEE Asia South Pacific Design Automation Conference, Yokohama, Japan, 24–27 January 2006, pp. 309–314.
- [22] D.B. Tuckerman, R.F.W. Pease, High-performance heat sinking for VLSI, IEEE Electron Device Letters EDL-2 (5) (1981).

- [23] J.M. Koo, S.J. Im, L. Jiang, K.E. Goodson, Integrated microchannel cooling for three-dimensional electronic circuit architectures, *Journal of Heat Transfer* (Transactions of the ASME) 127 (1) (2005) 49–58.
- [24] Y.J. Kim, et al., Thermal characterization of interlayer microfluidic cooling of three-dimensional integrated circuits with nonuniform heat flux, *Journal of Heat Transfer* (Transactions of the ASME) 132 (4) (2010) 1–9.
- [25] B.B. Dang, et al., Wafer-level microfluidic cooling interconnects for GSI, in: *Proceedings of the IEEE International Interconnect Technology Conference*, San Francisco, California, 6–8 June 2005, pp. 180–182.
- [26] A.K. Coskun, et al., Energy-efficient variable-flow liquid cooling in 3D stacked architectures, in: *Design, Automation and Test in Europe Conference*, Dresden, Germany, 8–12 March 2010, pp. 111–116.
- [27] Y.J. Lee, R. Goel, S.K. Lim, Multi-functional interconnect co-optimization for fast and reliable 3D stacked ICs, in: *Proceedings 2009 IEEE/ACM International Conference on Computer-Aided Design*, San Jose, California, 2–5 November 2009, pp. 645–651.
- [28] H. Mizunuma, C.-L. Yang, Y.-C. Lu, Thermal modeling for 3D-ICs with integrated microchannel cooling, in: *Proceedings of 2009 IEEE/ACM International Conference on Computer-Aided Design*, San Jose, California, 2–5 November 2009, pp. 256–263.
- [29] A. Sridhar, et al., 3D-ICE: Fast compact transient thermal modeling for 3D ICs with inter-tier liquid cooling, in: *Proceedings of 2010 IEEE/ACM International Conference on Computer-Aided Design*, San Jose, California, 7–11 November 2010, pp. 463–470.
- [30] A. Sridhar, et al., Compact transient thermal model for 3D ICs with liquid cooling via enhanced heat transfer cavity geometries, in: *Sixteenth International Workshop on Thermal Investigations of ICs and Systems (THERMINIC)*, Barcelona, Spain, October 2010, pp. 1–6.
- [31] C.B. Sobhan, S.V. Garimella, A comparative analysis of studies on heat transfer and fluid flow in microchannels, *Microscale Thermophysical Engineering* 5 (4) (2001) 293–311.
- [32] B. Amelifard, A. Afzali-Kusha, A. Khadernzadeh, Enhancing the efficiency of cluster voltage scaling technique for low-power application, in: *Proceedings of 2005 IEEE International Symposium on Circuits and Systems*, vol. 2, Kobe, Japan, 23–26 May 2005, pp. 1666–1669.
- [33] H. Qian, X. Huang, H. Yu, C.H. Chang, Cyber-physical thermal management of 3D multi-core cache processor system with microfluidic cooling, *ASP Journal of Low Power Electronics* 7 (1) (2011) 110–121.

- [34] D.A. Kaminski, M.K. Jensen, Introduction to Thermal and Fluids Engineering, John Wiley & Sons, Inc., 2005.
- [35] X.F. Peng, G.P. Peterson, Convective heat transfer and flow friction for water flow in microchannel structures, International Journal of Heat Mass Transfer 39 (12) (1996) 2599–2608.
- [36] E.N. Seider, G.E. Tate, Heat transfer and pressure drop of liquids in tubes, Industrial & Engineering Chemistry 28 (12) (1936) 1429–1435.
- [37] M.M. Sabry, A.K. Coskun, D. Atienza, Fuzzy control for enforcing energy efficiency in high-performance 3D systems, in: Proceedings of 2010 IEEE/ ACM International Conference on Computer-Aided Design, San Jose, California, 7–11 November 2010, pp. 642–648.
- [38] H. Oertel, et al., Prandtl-Essentials of Fluid Mechanics, Springer, 2009.
- [39] CurieJet Piezoelectric Micropump <<http://www.microjet.com.tw/>>.
- [40] ASCO miniature valves <<http://www.ascovalve.com/>>.
- [41] Wattch <<http://www.eecs.harvard.edu/dbrooks/wattch-form.html>>.
- [42] COMSOL Multiphysics <<http://www.comsol.com>>.
- [43] T.A. Davis, E. Palamadai, KLU: Sparse LU Factorization Algorithm for Circuit Simulation <<http://www.cise.ufl.edu/research/sparse/klu/>>.
- [44] T.A. Davis, SuiteSparse: Collection of Packages for Sparse Matrix Algorithms <<http://www.cise.ufl.edu/research/sparse/SuiteSparse/>>.
- [45] SPEC2000: Standard Performance Evaluation Corporation <<http://www.spec.org/cpu2000/>>.
- [46] HotSpot <<http://lava.cs.virginia.edu/HotSpot/>>.

Hanhua Qian received his B. Eng. (Hons) degree from Nanyang Technological University in Singapore in 2008. He was working as a product design engineer in industry in 2008. From January 2009, he has been working towards his Ph. D. degree in School of Electrical and Electronic Engineering at Nanyang Technological University Singapore. His main research interest includes 3DIC design, digital filter design and Software Defined Radio (SDR).

Chip-Hong Chang received his B. Eng. (Hons.) degree from the National University of Singapore, Singapore, in 1989, and the M. Eng. and Ph. D. degrees from Nanyang Technological University (NTU), Singapore, in 1993 and 1998, respectively. He served as a Technical Consultant in industry prior to joining the School of Electrical and Electronic Engineering (EEE), NTU, in 1999, where he is currently an Associate Professor. He holds joint appointments with the university as Assistant Chair of Alumni of the School of EEE since June 2008, Deputy Director of the Center for High Performance Embedded Systems from 2000 to 2011, and the Program Director of the Center for Integrated Circuits and Systems from 2003 to 2009. He has co-edited one book, published three book chapters and more than 170 research papers in refereed international journals and conferences. His current research interests include low power arithmetic circuits, residue number system, digital filter design, and digital watermarking for VLSI intellectual property protection.

Dr. Chang serves as Associate Editor of the IEEE Transactions on Circuits and Systems—I in 2010–2011 and 2012–2013 and the IEEE Transactions on Very Large Scale Integration (VLSI) Systems since 2011, and the Guest Editor of special issues in the Journal of Circuits, Systems and Computers in 2011, IEEE Transactions on Circuits and Systems—I and Journal of Electrical and Computer Engineering in 2012. He is the Editorial Advisory Board Member of the Open Electrical and Electronic Engineering Journal since 2007 and the Editorial Board Member of the Journal of Electrical and Computer Engineering since 2008. He has also served in many international conference advisory and technical program committees. He is a Fellow of the IET and a senior member of the IEEE.

Hao Yu obtained his B.S. degree from Fudan University (Shanghai China) in 1999, and M.S./Ph. D degrees both from Electrical Engineering Department at UCLA in 2007, with major of the integrated circuit and embedded computing. He was a senior research staff at Berkeley Design Automation (BDA). Since October 2009, he is an Assistant Professor at School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore. His primary research interests are 3D computing system and designs at nano-tera scale. He has 52 major peer-reviewed publications, best paper award in ACM Transactions on Design Automation of Electronic Systems (TODAES), best paper award nominations in DAC/ICCAD/ASP-DAC, and inventor award from semiconductor research cooperation (SRC). He is the Associate Editor and technical program committee member of several journals and conferences.

List of Tables

Table 1	Experiment settings.
Table 2	Comparison of average temperature of each row in Fig. 5.
Table 3	Comparison of max and min temperatures between proposed thermal model and COMSOL.
Table 4	Clustering results with different η values.
Table 5	Reallocation of cooling resources.
Table 6	Comparison of the maximum temperatures under uniform and clustered coolings.
Table 7	Maximum temperature after minimization of cooling effort.
Table 8	Comparison of cooling effort under thermal constraint.

List of Figures

- Fig. 1 Schematic of 3D integrated circuit with microfluidic cooling.
- Fig. 2 Grid division of 3D integrated circuit.
- Fig. 3 Demonstration of decaying Nusselt number along flow direction.
- Fig. 4 An illustration of non-uniform power distribution (W/cm^2) of 3DIC. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
- Fig. 5 Comparison of top layer temperature profile against COMSOL at a heat flux of $150 \text{ W}/\text{cm}^2$ and $1 \text{ ml}/\text{min}$ flow rate per channel, coolant flows from north (upper) to south (lower) direction. (a) COMSOL. (b) Proposed thermal model.
- Fig. 6 More accurate minimum temperature prediction with fine-grained grid division.
- Fig. 7 Cooling demand of microchannels with different power distributions. (a) Case 1. (b) Case 2. (c) Case 3. (d) Case 4. (e) Case 5.
- Fig. 8 Temperature distribution comparison: thermal maps of the top layer for Case 1 under uniform flow rate cooling and reallocated clustered flow rate cooling with the same total flow rate, coolant flows from north (upper) to south (lower) direction. (a) Uniform flow rate cooling. (b) Clustered cooling effort.
- Fig. 9 Temperature distribution comparison: top layer thermal maps of Case 1 using the proposed minimization method and uniform flow rate minimization with the same thermal limit, coolant flows from north

(upper) to south (lower) direction. (a) Proposed minimization method. (b)
Uniform flow rate minimization.

Parameter	2-Layer stack	4-Layer stack
Chip size (mm)	1.8×1.8	32×32
Problem size	405	37 888
Default total flow rate	3.6–9 ml/min	288 ml/min
BEOL layer thickness		12 μm
Device layer thickness		2 μm
Bulk thickness		48 μm
Microchannel height		100 μm
Microchannel width		100 μm
Channel wall width		100 μm
Ambient temperature		300 K (26.85 °C)
Thermal threshold		358.15 K (85 °C)
Coolant type		Deionized water

Table 1

Row number	1	2	3	4	5	6	7	8	9
COMSOL	310.37	320.02	326.75	331.57	335.24	338.06	340.20	341.72	342.54
Our Model	307.18	312.92	319.09	324.89	330.05	334.44	337.96	340.48	341.86
Error (%)	1.03	2.22	2.34	2.01	1.55	1.07	0.66	0.36	0.20

Table 2

Heat flux (w/cm ²)	Flow rate per microchannel								Error (%)
	1 ml/min		0.8 ml/min		0.6 ml/min		0.4 ml/min		
	COMSOL	Ours	COMSOL	Ours	COMSOL	Ours	COMSOL	Ours	
150	342.91	341.86	346.73	347.09	351.82	355.10	360.15	369.68	0.84
	305.16	307.17	305.54	308.87	306.04	311.88	306.81	318.49	1.87
100	328.61	327.91	331.15	331.39	334.55	336.73	340.10	346.45	0.59
	303.44	304.78	303.70	305.91	304.03	307.92	304.54	312.32	1.23
50	314.30	313.95	315.58	315.70	317.27	318.37	320.05	323.23	0.32
	301.72	302.39	301.85	302.96	302.01	303.96	302.27	306.16	0.63
10	302.86	302.79	303.12	303.14	303.45	303.67	304.01	304.65	0.07
	300.34	300.48	300.37	300.59	300.40	300.79	300.45	301.23	0.13

Table 3

Case no.	η (%)	N	Number of channels per cluster
1	20	3	300, 60, 120
	10	5	300, 60, 20, 60, 40
	5	7	280, 20, 60, 20, 60, 20, 20
2	20	3	360, 100, 20
	10	6	300, 60, 20, 50, 30, 20
	5	9	40, 200, 60, 40, 20, 20, 50, 30, 20
3	20	3	360, 100, 20
	10	6	300, 60, 20, 50, 30, 20
	5	9	40, 200, 60, 40, 20, 20, 50, 30, 20
4	20	3	360, 100, 20
	10	6	200, 100, 60, 40, 60, 20
	5	10	200, 100, 40, 20, 20, 10, 25, 25, 20
5	20	2	72, 24
	10	4	300, 60, 40, 80
	5	6	100, 300, 60, 40, 20, 60

Table 4

Case no.	<i>N</i>	Flow rate per channel (ml/min)
1	3	0.348, 0.798, 1.131
2	3	0.417, 1.126, 1.255
3	3	0.417, 1.128, 1.250
4	3	0.412, 1.139, 1.281
5	4	0.403, 0.696, 0.996, 1.068

Table 5

Case no.	Uniform (K)	Clustered (Case 5) (K)	δT (K)
1	333.62	326.40 (327.48)	6.68
2	332.02	326.64 (326.02)	5.69
3	333.63	327.45 (327.50)	6.16
4	333.94	327.49 (327.68)	6.36

Table 6

Case no.	φ (Case 5)	Max temperature (Case 5) (K)
1	0.35 (0.38)	354.06 (352.84)
2	0.37 (0.36)	352.52 (352.16)
3	0.36 (0.38)	354.30 (352.72)
4	0.34 (0.36)	356.00 (353.99)

Table 7

Case no.	1	2	3	4
T_{max} (K)				
Uniform	357.28	357.73	357.27	358.01
Cluster	357.85	357.43	358.01	358.10
Case 5	357.69	358.40	357.55	357.62
Q_{total}				
Uniform	119.95	112.58	120.01	117.72
Cluster	91.93	94.70	94.97	93.42
Case 5	95.69	90.24	96.02	95.03
Flow rate saving (%)	21.8	17.9	20.4	19.96

Table 8

Fig. 1

Fig. 2

Fig. 3

Fig. 4

Fig. 5

Fig. 6

Fig. 7

Fig. 8

Fig. 9