Whitespace Redistribution For Thermal Via Insertion In 3D Stacked ICs*

Eric Wong and Sung Kyu Lim
School of Electrical and Computer Engineering
Georgia Institute of Technology
limsk@ece.gatech.edu

Abstract

One of the biggest challenges in 3D stacked IC design is heat dissipation. Incorporating thermal vias is a promising method for reducing the temperatures of 3D ICs. The bonding styles between device layers impose certain restrictions to where thermal vias may be inserted. This paper presents a whitespace redistribution algorithm that takes bonding style into consideration to improve thermal via placement, which in turn reduces temperature.

1 Introduction

Three dimensional (3D) integrated circuits are an emerging technology with great potential to improve performance. The ability to route signals in the vertical dimension enables distant blocks to be placed on top of each other. This results in a decrease in the overall wire length, which translates into less wire delay and greater performance. The wafer-bonding approach in [1] joins discrete wafers using a copper interconnect interface, and permits multiple wafers with multiple 3D interconnects. Wafers can be bonded with metal routing layers facing each other (face-to-face), with the substrates facing each other (back-to-back), or with metal layer facing substrate (face-to-back) as illustrated in Figure 1.

One of the biggest challenges of 3D circuit design is heat dissipation. In 3D circuits, more devices are packed into a smaller area, resulting in higher power densities. In addition, heat from the core of a 3D chip has to travel through layers of low conductivity dielectric, inserted between device layers, to reach a heat sink. One method for mitigating the thermal issue is to insert thermal vias that are used to establish thermal paths from the core of a chip to the outer layers. A few recent works have considered thermal vias in 3D ICs [2, 3, 4, 5].

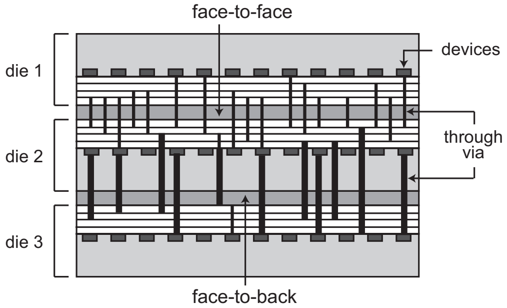


Figure 1. Through vias in 3D ICs with face-to-face and face-to-back bonding. Back-to-back style forms when the two substrate sides are attached (not shown in this figure).

Most floorplanning approaches compact all of the blocks to the lower left corner. Although this gives optimal area, it is not necessarily optimal for wire length, temperature or other objectives. Whitespace redistribution for wire length minimization was formulated as a linear program(LP) in [6] and solved using network flow. This paper presents a LP based whitespace redistribution algorithm for the purpose of reducing temperature. Our whitespace redistribution algorithm also takes the effect of bonding styles on thermal via insertion into account.

2 Problem Formulation

The following are given as the input to the Thermal Via Aware 3D Whitespace Redistribution Problem: (i) a set of blocks, (ii) the width, height, and average power density of each block, (iii) a net list that specifies the connections between blocks, (iv) the number of device layers in the 3D IC, (v) the bonding styles of adjacent layers, (vi) a floorplan of the blocks, and (vii) A_{tot} the footprint area of the chip. The goal of the Thermal Via Aware 3D Whitespace Redistribution Problem is to modify the location of each block in the floorplan and to plan the locations of thermal

*This material is based upon work supported by the National Science Foundation under Grant No. 0546382, MARCO C2S2, and MARCO IFC.

1-4244-1258-7/07/$25.00 ©2007 IEEE

vias, to minimize the chip temperature, without increasing A_{tot}, the footprint area of the chip. Thermal vias may be inserted anywhere for F2F bonded layers. A F2B bond requires whitespace to be available in the layer with its substrate side bonded in order to insert thermal vias. For B2B bonding both layers must have whitespace that align with each other in order to insert thermal vias.

3 Overview of Algorithm

Simulated annealing is a very popular approach for floorplanning due to its high quality solutions and flexibility in handling non-linear objectives. Simulated annealing begins with an initial floorplan and its cost in terms of area, wirelength, and maximum chip temperature. Then a random perturbation (move) is made to the initial solution to generate a new floorplan and its new cost is calculated. If the new cost is lower than the old one, then the new floorplan is accepted; otherwise there is a probability of acceptance based on the temperature of the annealing schedule. The higher the annealing temperature, the higher the acceptance probability. At each annealing temperature level a predetermined number of floorplans are examined. The annealing temperature is decreased exponentially, and the annealing process terminates when the freezing temperature is reached. To represent non-slicing floorplans, sequence pair [7] is used. In order to extend sequence pair to 3D, one sequence pair was used for each layer. The same was done to extend normalized polish expressions to 3D.

After floorplanning, whitespace redistribution is performed. First, thermal analysis is performed. Next, thermal tiles near hot spots are assigned a repelling factor based on temperature. Then blocks are moved away from thermal tiles with higher repelling factors. After moving blocks, thermal analysis is performed again. If the thermal profile changes significantly, repelling factors are assigned and blocks are moved again. Iteration between thermal analysis and module movement is done until the temperature profile stops improving.

3.1 Thermal Model

A set of 3D thermal tiles are used for thermal analysis. Each thermal tile models a small volume of the 3D die stack. Each thermal tile is connected to adjacent tiles by thermal resistors as shown in Figure 2. This is equivalent to using a discrete approximation of the steady state thermal equation $-k\nabla^2 T = P$, where k is thermal conductivity, T is temperature, and P is power. This results in the matrix equation $G \cdot t = p$, where G is a thermal conductivity matrix, p is a power vector, and t is a temperature vector. One way to solve this matrix equation would be to invert the matrix $G^{-1} = R$, which takes $O(n^3)$ time. Then t can be calculated through matrix multiplication $t = R \cdot p$, which takes $O(n^2)$ time.

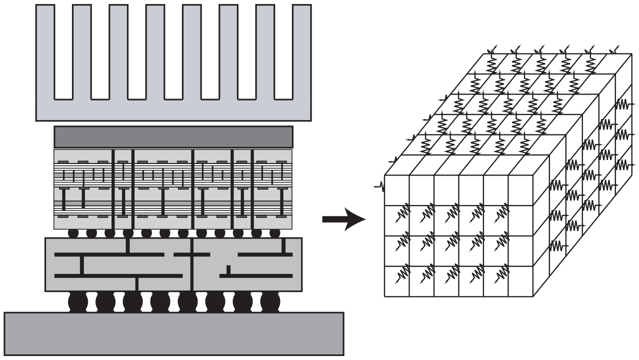


Figure 2. Thermal modelling.

During thermal driven floorplanning, moving blocks around does not significantly change the thermal conductivities. The power profile changes are mainly responsible for the changes in temperature. This allows the G matrix to be inverted to R once in the beginning and reused for subsequent temperature calculations. Only the power vector needs to be changed, so temperature calculations only require one matrix multiplication. This allows the temperature of each floorplan to be evaluated in $O(n^2)$ time rather than $O(n^3)$ time.

3.2 Whitespace Redistribution

The goal of whitespace redistribution is to move blocks away from hot spots to make space for thermal vias. This can also reduce temperature by reducing the power density near the hot spots. The movement of blocks away from hot spots is formulated as a linear program.

Given:

- W_{max} = width of the floorplan
- H_{max} = height of the floorplan
- N = set of blocks
- M = set of thermal tiles
- x_j, y_j = coordinates of the center of thermal tile j
- r_j = repelling factor of thermal tile j
- w_i = width of block i
- h_i = height of block i
- G_h = horizontal constraint graph of the floorplan
- G_v = vertical constraint graph of the floorplan

Maximize:

$$\sum_{j \in M} r_j \cdot c_j$$

Subject to:

$$a_{ij} = abs(x_i - x_j) \qquad i \in N, j \in M \qquad (1)$$

$$b_{ij} = abs(y_i - y_j) \qquad i \in N, j \in M \qquad (2)$$

$$c_j \leq a_{ij} + b_{ij} \qquad i \in N, j \in M \qquad (3)$$

$$x_i - 0.5w_i \geq 0 \qquad i \in N \qquad (4)$$

$$y_i - 0.5h_i \geq 0 \qquad i \in N \qquad (5)$$

$$x_i + 0.5w_i \leq W_{max} \qquad i \in N \qquad (6)$$

$$y_i + 0.5h_i \leq H_{max} \qquad i \in N \qquad (7)$$

$$x_l + 0.5w_l \leq x_r - 0.5w_r \qquad e(l,r) \in G_h \qquad (8)$$

$$y_d + 0.5h_d \leq y_u - 0.5h_u \qquad e(d,u) \in G_v \qquad (9)$$

Where x_i and y_i are the coordinates of the center of block i, a_{ij} is the distance between the centers of block i and thermal tile j in the x-direction, b_{ij} is the distance between the centers of block i and thermal tile j in the y-direction, and c_j is the manhattan distance between thermal tile j and its closest block. The objective function is to maximize the manhattan distance between each thermal tile and its closest block weighted by the repelling factor of each thermal tile. Constraints 1 and 2 determine the x and y distances between each block and thermal tile. For each thermal tile, constraint 3 determines the manhattan distance of the closest block. Constraints 4 through 7 ensure that the blocks stay within the footprint of the floorplan. Constraints 8 and 9 prevent blocks from overlapping by enforcing the horizontal and vertical constraint graphs of the floorplan.

Horizontal and vertical constraint graphs can be constructed for may different types of floorplan representations. For a floorplan based on sequence pair (Γ_p, Γ_m), horizontal and vertical constraint graphs G_h and G_v can be constructed based on the following relationships.

$(\langle \cdots b_i \cdots b_j \cdots \rangle, \langle \cdots b_i \cdots b_j \cdots \rangle) \Rightarrow b_i$ is to the left of b_j

$(\langle \cdots b_j \cdots b_i \cdots \rangle, \langle \cdots b_i \cdots b_j \cdots \rangle) \Rightarrow b_i$ is below b_j

For the first relationship an edge from block b_i to block b_j would be added to G_h, and for the second relationship an edge from block b_i to block b_j would be added to G_v.

The repelling factor r_j of each thermal tile is calculated from temperature and thermal via demand. To calculate the repelling factor the temperature profile is normalized, and then the normalized repelling factors are cubed. This gives thermal tiles with high temperature much higher priority for repelling blocks than thermal tiles with low temperature. In order to take back-to-back thermal vias into account, a thermal tile with a high repelling factor on a back-to-back layer will increase the repelling factor of its aligned thermal tile on the corresponding layer. This increases the chance that there will be whitespace alignment for the two back-to-back layers, allowing for thermal via insertion.

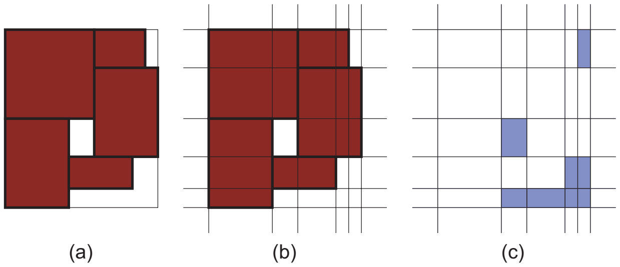


Figure 3. Whitespace detection. (a) starting floorplan, (b) non-uniform grid lines each correspond to a boundary of a block, (c) whitespaces detected

Whitespace redistribution changes the locations of blocks. This changes the thermal profile of a floorplan and may modify the locations of hot spots. For this reason, iteration between thermal analysis and block movement is performed until the temperature stops decreasing.

3.3 Via Insertion

Thermal via insertion in face-to-back and back-to-back bonded layers are constrained by available whitespace, so whitespace detection must be performed. This can be done by drawing a grid on the floorplan and marking the grid cells that are covered by blocks. The grid cells that remain unmarked are detected as whitespace. If the grid is too coarse, the sizes and locations of the whitespaces will be inaccurate. In order to get the exact locations and sizes of the whitespaces, we use a non-uniform grid as shown in Figure. 3b. Vertical grid lines are drawn at the left and right edges of each block. Horizontal grid lines are drawn at the top and bottom edges of each block. If there are n blocks, there can be up to $2n$ vertical and $2n$ horizontal grid lines. The maximum number of grid cells is $(2n-1)^2 = 4n^2 - 4n + 1$ making this whitespace detection algorithm $O(n^2)$. After whitespace detection is performed, the maximum number of vias that can be inserted is calculated based on the bonding styles and the available whitespace. Once the vias are added, the thermal resistor values between thermal tiles are updated and thermal analysis can be performed.

Table 1. Thermal driven whitespace redistribution versus wire length driven whitespace redistribution.

	Thermal driven redistribution					Wire length driven redistribution [6]			
						FAST-SP		Parquet	
	area	wire length		temperature		wire length		wire length	
ckts		original	after	original	after	original	after	original	after
apte	52.2	440.3	449.7	86	83	426.7	418.9	476.3	458.0
xerox	19.8	542.5	555.2	86	85	486.1	462.8	581.6	550.6
hp	12.0	181.2	170.1	100	99	170.0	161.5	161.1	152.4
ami33	1.3	75.0	74.3	101	92	60.0	58.0	77.2	74.1
ami49	41.8	926.7	949.3	63	61	790.1	760.2	857.8	818.9
Ratio	-	1.000	0.999	1.000	0.965	1.000	0.963	1.000	0.954

Table 2. Whitespace redistribution versus thermal driven floorplanning.

	Before redistribution			After redistribution			CBA [8]			CBA-T [8]		
		wire			wire			wire			wire	
ckts	area	length	temp	area	length	temp	area	length	temp	area	length	temp
ami33	359856	23408	201	359856	25452	184	353000	22495	471	414000	24442	160
ami49	11143000	634494	252	11143000	657646	233	14900000	446815	259	18400000	477646	151
n100	51870	72120	398	51870	74258	253	52900	100525	391	65600	92450	158
n200	51675	141013	261	51675	144578	213	57700	210333	323	69100	190886	156
n300	80384	219184	332	80384	223007	317	89000	314980	373	106000	253837	167
Ratio	1.000	1.000	1.000	1.000	1.039	0.849	1.000	1.000	1.000	1.207	0.958	0.451

4 Experimental Results

The algorithms in the paper were implemented in C++. The experiments were run on Pentium IV 2.4 Ghz dual processor systems running linux. Ten GSRC benchmarks were used. The blocks were randomly assigned power densities between $10^6\ W/m^2$ and $5 \cdot 10^6\ W/m^2$. The 3D floorplans have four placement layers with face-to-face, back-to-back, and face-to-face bonding.

Table 1 compares our thermal driven whitespace redistribution to the wire length driven whitespace redistribution from [6] based on the MCNC benchmarks. Our thermal driven whitespace redistribution algorithm was able to reduce the temperature of all of the benchmarks. As a side effect three of the benchmarks had increased wire length while the other two benchmarks had a decrease in wire length. The wirelength driven whitespace redistribution algorithm from [6] was able to reduce the wire lengths of all of the benchmarks.

Table 2 compares whitespace redistribution with another method for reducing temperature, thermal driven floorplanning [8]. Thermal vias were not used in this comparison. Relative to area and wire length driven floorplanning (CBA), thermal driven floorplanning (CBA-T) achieved a 55% reduction in temperature at a cost of 20% area increase. Our whitespace redistribution algorithm achieved a more modest 15% reduction in temperature, but no area expansion was required.

Table 3 shows the effect of our whitespace redistribution algorithm on 3D floorplans. Whitespace redistribution reduced temperature by over 8% at a cost of approximately 3% wire length increase. In six of the cases, whitespace redistribution was able to increase the amount of whitespace overlap near hot spots in the B2B bonded layers. Even in the cases where whitespace overlap was not increased, the temperature was reduced due to moving blocks away from hot spots. Table 4 shows the effect of whitespace redistribution on 2D floorplans where thermal via insertion is not possible. In this situation, whitespace redistribution reduced the temperature by over 7% with less than a 1% wire length increase.

The objective of our 3D floorplanner so far was area and wire length. Table 5 shows the effect of whitespace redistribution on thermal driven floorplanning. The temperature reduction was over 5% at a cost of less than 5% wire length increase. Whitespace redistribution may not be as efficient for thermal driven floorplanning, but the final temperature (after redistribution with thermal vias) is lower for thermal driven floorplanning than area and wire length driven floorplanning. Figure 4 show a non-slicing 3D floorplan of n50c before and after whitespace redistribution. In the fourth layer, several blocks moved down which allowed more thermal vias to be inserted near the hottest blocks.

Table 3. Area and wire length driven 3D floorplanning with whitespace redistribution.

		before redistribution				after redistribution					
ckts	area	wire length	temp w/o vias	temp w/ vias	whitespace overlap	wire length	temp w/o vias	temp w/ vias	whitespace overlap	redist time	total runtime
n50	59085	46003	411	300	0.0008	48399	364	261	0.0477	314	570
n50b	58016	41254	239	187	0.0000	43322	208	168	0.0000	224	488
n50c	59616	44399	305	243	0.0000	45998	290	229	0.0561	327	565
n100	51870	72120	398	298	0.0014	74258	330	250	0.0247	3346	4734
n100b	47874	67489	240	188	0.0005	70377	212	167	0.0153	4980	5937
n100c	51597	73460	272	210	0.0000	76061	241	190	0.0000	4865	5772
n200	51675	141013	261	212	0.0000	144578	253	203	0.0129	16924	19549
n200b	52459	149452	261	209	0.0000	152428	245	198	0.0000	19442	20949
n200c	51490	149492	222	180	0.0000	153976	213	174	0.0000	24493	26481
Ratio	-	1.000	1.000	1.000	-	1.034	0.913	0.917	-	0.762	1.000

Table 4. Area and wire length driven 2D floorplanning with whitespace redistribution.

		before redistribution		after redistribution			
ckts	area	wire length	temp	wire length	temp	redist time	total runtime
n50	224028	99303	177	100188	165	468	829
n50b	233730	87562	122	88551	106	463	811
n50c	233288	95722	172	95851	145	433	793
n100	209040	169997	76	171340	69	7029	9107
n100b	184005	149352	67	151540	60	7044	9156
n100c	198468	159314	76	160658	75	6989	9182
n200	195596	322952	59	323956	53	21802	27631
n200b	200220	280691	95	281869	93	27566	32722
n200c	191619	303798	55	304643	54	28330	34629
Ratio	-	1.000	1.000	1.007	0.927	0.732	1.000

5 Conclusions

A LP based whitespace redistribution algorithm for bonding style aware thermal via planning was presented in this paper. Experimental results show that whitespace redistribution can significantly reduce temperatures in 3D ICs. When combined with thermal driven floorplanning, chip temperature was even lower. With minor modifications, the whitespace redistribution algorithm also reduced temperatures in traditional 2D ICs.

References

[1] A. Fan, A. Rahman, and R. Reif, "Copper wafer bonding," *Electrochemical Solid-State Letter*, 1999.

[2] J. Cong and Y. Zhang, "Thermal via planning for 3-d ics," in *Proc. IEEE Int. Conf. on Computer-Aided Design*, 2005.

[3] Z. Li, X. Hong, Q. Zhao, S. Zeng, J. Bian, H. Yang, and C. K. Cheng, "Integrating dynamic thermal via planning with 3d floorplanning algorithm," in *Proc. Int. Symp. on Physical Design*, 2006.

[4] E. Wong and S. K. Lim, "3d floorplanning with thermal vias," in *Proc. Design, Automation and Test in Europe*, 2006, pp. 878–883.

[5] B. Goplen and S. Sapatnekar, "Thermal Via Placement in 3-D ICs," in *Proc. Int. Symp. on Physical Design*, 2005.

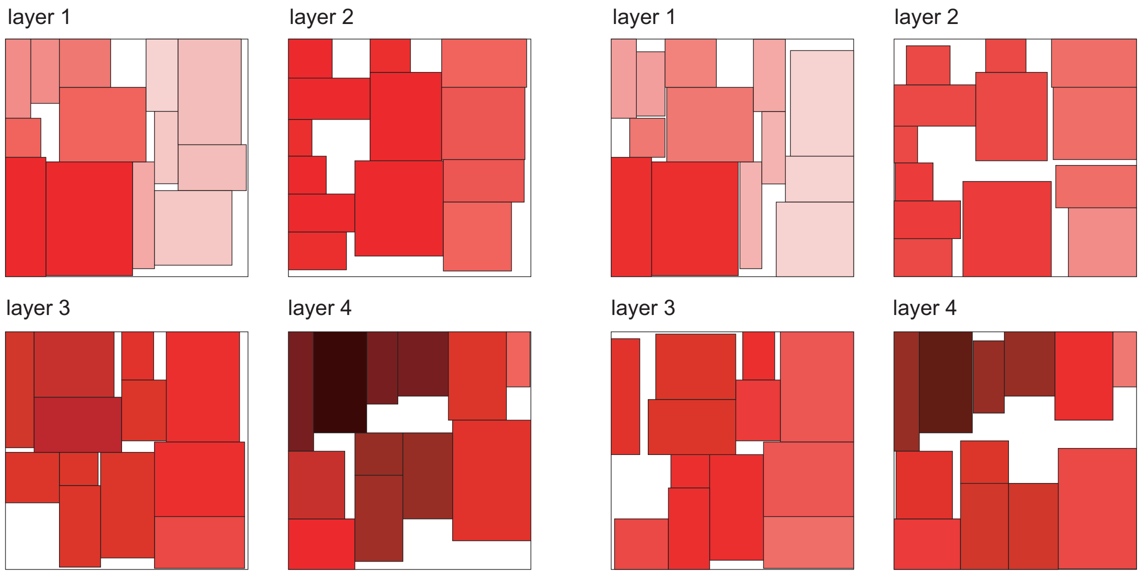
[6] X. Tang, R. Tian, and D. F. Wong, "Optimal redistribution of white space for wire length minimization," in *Proc. Asia and South Pacific Design Automation Conf.*, 2005, pp. 412–417.

[7] H. Murata, K. Fujiyoshi, S. Nakatake, and Y. Kajitani, "Rectangle packing based module placement," in *Proc. IEEE Int. Conf. on Computer-Aided Design*, 1995, pp. 472–479.

[8] J. Cong, J. Wei, and Y. Zhang, "A Thermal-Driven Floorplanning Algorithm for 3D ICs," in *Proc. IEEE Int. Conf. on Computer-Aided Design*, 2004.

Table 5. Thermal driven 3D floorplanning with whitespace redistribution.

		without redistribution				with redistribution					
ckts	area	wire length	temp w/o vias	temp w/ vias	whitespace overlap	wire length	temp w/o vias	temp w/ vias	whitespace overlap	redist time	total runtime
n50	58140	41985	248	208	0.0000	44460	230	196	0.0000	337	1152
n50b	61275	45004	211	180	0.0000	46391	202	169	0.0088	953	923
n50c	63392	43016	198	177	0.0088	46644	178	150	0.0263	1015	987
n100	55584	73265	185	156	0.0086	77189	176	144	0.0000	7565	6980
n100b	49046	72878	151	123	0.0000	75581	143	116	0.0000	8442	6923
n100c	53436	74486	154	122	0.0042	78069	145	115	0.0042	7714	6526
n200	57600	150945	123	103	0.0000	155963	119	100	0.0000	28737	36740
n200b	55752	128780	135	112	0.0331	133554	130	109	0.0827	25093	32734
n200c	54902	146694	132	110	0.0118	154128	129	107	0.0000	28749	36689
Ratio	-	1.000	1.000	1.000	-	1.048	0.948	0.939	-	0.535	1.000



before redistribution

after redistribution

Figure 4. A non-slicing 3D floorplan of n50c before and after whitespace redistribution. Darker shading indicates higher temperature.