TILE-BASED POWER PLANNING DURING FLOORPLANNING

Jyh Perng Fang and Sao Jie Chen

Graduate Institute of Electronics Engineering and
Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan, ROC
E-mail: jpfang@ntut.edu.tw and csj@cc.ee.ntu.edu.tw

ABSTRACT

In this paper, we introduce a tile-based approach to power planning at the stage of floorplanning. For a given flooplan solution, an associated tile graph of power density is generated, the temperature of the floorplan is evaluated tile by tile. In contrast to the direct evaluation from the power consumption of circuit blocks and neglect the effect of heat diffusion, we take the effect of heat diffusion in a die into consideration. Also, we simplify the computing of temperature by way of a tile graph, which make the heat estimation and thus the power planning in floorplanning stage possible.

1. INTRODUCTION

With the fast progress of modern VLSI technology, the chip density and the clock speed have been highly raised. The higher clock speed causes larger dynamic power consumption, which may drastically raise the temperature at some regions on a die, and form the so-called "hot-spots". These hot spots have significant effect not only on the chip performance but also on the chip reliability. Therefore, the temperature problem is becoming a critical issue of contemporary VLSI design. To remove the hot spots of a design, adequate power planning is necessary. With an effective power planning, the power dissipation can be distributed over the whole chip more evenly rather than concentrated at some specific regions.

Traditionally, floorplanning is used to plan the location and aspect ratio of modules in a chip [1,2]. To decrease design iteration, the power planning which was conventionally processed during later design stage is now considered at floorplanning [3,4]. However, those works consider power delivering problem only, the problem of heat diffusion is not dealt.

On the other hand, there have been more and more works concentrating on the temperature problem of a die. For example, Cheng and Kang [5] proposed an efficient approach to identify hot-spot on a die. Ajami et al. [6] analyzed and modeled the effects of substrate temperature on the signal integrity.

Following the footprint of these previous researches, we introduce a tile-based approach to power planning at the floorplanning stage. We map the floorplan into a tile graph of power density and estimate the temperature rise of each tile, and then, through an efficient approach, the

hot tiles on the tile graph are promptly identified.

To show the efficiency and effectiveness of our approach, we integrate it into a B*-tree floorplanner, and the experimental results prove that, through our approach, power planning in terms of heat analysis can be evaluated very fast. Thus, at the stage of floorplanning, we can evaluate the temperature of each floorplan solution rather than identifying the hot spots of a final floorplan

This paper is organized as follows. Sections 2 and 3 respectively introduce the preliminaries and the problem formulation. Section 4 describes the concept of hot tile and the approach to identify it. The experimental results are presented in Section 5. The last section draws a conclusion.

2. PRELIMINARIES

In the work of [5], a method of thermal analysis for early identification of hot-spot in a chip is introduced. In [5], the heat diffusion on a chip is deduced as an integral of product of error functions,

$$\Delta T(x,y,0,t) = \frac{\alpha P}{4k(abc)} \int_0^t [erf(\frac{A_1}{4\sqrt{\alpha\tau}}) + erf(\frac{A_2}{4\sqrt{\alpha\tau}})] \times$$

$$[erf(\frac{B_1}{4\sqrt{\alpha\tau}}) + erf(\frac{B_2}{4\sqrt{\alpha\tau}})] \times erf(\frac{C}{4\sqrt{\alpha\tau}}) d\tau, \ldots\ldots\ldots(1)$$

in which, $\Delta T(x,y,0,t)$ represents the temperature raised on an observation point located at chip surface with a Manhattan distance of $x+y$ from heat source after time t; α is the thermal diffusivity; P is the power dissipated by a heat source; k is an adjustable parameter; a, b, c are respectively the length, width, thickness of the heat source; A_1, A_2, B_1, B_2, C are defined as $A_1=2(a/2+x)$, $A_2=2(a/2-x)$, $B_1=2(b/2+y)$, $B_2=2(b/2-y)$, $C=2c$, respectively. To solve Expression (1) analytically, the error function is approximated by a piecewise linear function [5].

To simplify the computation needed in solving Expression (1), we treat the heat source as the origin point and let the observation point be constrained at the first quadrant and y be constrained smaller than x. Finally, based on these approximations and constraints, eight solutions are deduced from Expression (1). For reference, these cases are duplicated as shown in Fig. 2.1, in which $t_{a1}=A_1^2/(4\pi\alpha)$, $t_{a2}=A_2^2/(4\pi\alpha)$, $t_{b1}=B_1^2/(4\pi\alpha)$, $t_{b2}=B_2^2/(4\pi\alpha)$, and $t_c=C^2/(4\pi\alpha)$.

$$1.\ t_{a1} \geq t_{b1} \geq t_{b2} \geq t_{a2} \geq t_c \quad 2.\ t_{a1} \geq t_{b1} \geq t_{a2} \geq t_{b2} \geq t_c$$
$$3.\ t_{a1} \geq t_{a2} \geq t_{b1} \geq t_{b2} \geq t_c \quad 4.\ t_{a1} \geq t_{a2} \geq t_{b1} \geq t_c \geq t_{b2}$$
$$5.\ t_{a1} \geq t_{b1} \geq t_{a2} \geq t_c \geq t_{b2} \quad 6.\ t_{a1} \geq t_{b1} \geq t_{b2} \geq t_c \geq t_{a2}$$
$$7.\ t_{a1} \geq t_{b1} \geq t_c \geq t_{b2} \geq t_{a2} \quad 8.\ t_{a1} \geq t_{b1} \geq t_c \geq t_{a2} \geq t_{b2}$$

Figure 2.1 Eight cases for the solution of Expression

The solutions for these cases are somewhat complicated. For example, the solution under Case 7 is:

$$\Delta T(x,y,0,\infty) = \frac{\alpha P}{4k(abc)}\left[t_{b2} - t_c + (\sqrt{t_{a2}}\sqrt{t_{b2}})(2.5 + \right.$$

$$\log\frac{t_c}{t_{b2}}) + (\sqrt{t_{a2}}\sqrt{t_c})(-2.5 + \log\frac{t_c}{t_{b1}}) + (\sqrt{t_{b2}}\sqrt{t_c})(-2.5$$

$$\left. + \log\frac{t_c}{t_{a1}}) + (\sqrt{t_{b1}}\sqrt{t_c})(2.5 + \log\frac{t_{a1}}{t_{b1}})\right] \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(2)$$

As a heat simulation technique, the method introduced by [5] is efficient compared to other numerical methods, for example, that in [6]. However, during floorplaning, the computation of accumulative temperature using Expression (2) for all circuit blocks will substantially lengthen the computation time of the iterative floorplanning algorithm, because each circuit block has its own power dissipation and will influence the temperature of every other circuit blocks. Obviously, to analyze the temperature raised by power dissipation during floorplanning, we need a more efficient method.

3. PROBLEM FORMULATION

The problem is formulated as follows:

Given: A floorplan f consisting of a set of circuit blocks $B=\{b_1, b_2, \ldots, b_n\}$, a set of power consumptions $P=\{p_1, p_2, \ldots, p_n\}$ associated to the set of block.

Goal: Find a floorplan solution such that a threshold value of temperature rise is not violated and the area is optimized.

Some terminologies are defined in the following.

Tile Graph: A tile graph $G(V,E)$ contains a set of tiles V and a set of edges E. For any two neighboring tiles u and v, there exists an edge $e_{u,v}$ connecting these two tiles. Each tile v has a weighting $w(v)$, and each edge has a weighting $w(e_{u,v})=1$.

Tile Distance (TD): For any two tiles on a tile graph, the tile distance is the number of edges between them. The x component and y component of TD are defined as TD_x and TD_y, respectively.

4. HOT TILES

In this section, the procedure to build the tile graph and an efficient approach extending and strengthening the method of [5] for identifying hot tile is introduced.

Instead of treating circuit block as a heat source, we treat tile as a heat source, thereby, the tile of heat source and the tile at observation point are termed heat source tile and target tile, respectively.

For a given floorplan f, we can build a tile graph $G(V,E)$

of power density according to the power specification of each circuit block and the spatial relationship between circuit blocks. The tile graph corresponding to Fig. 4.1(a) is represented as a 2-D array, as shown in Fig. 4.1(b), in which all tiles have equal size.

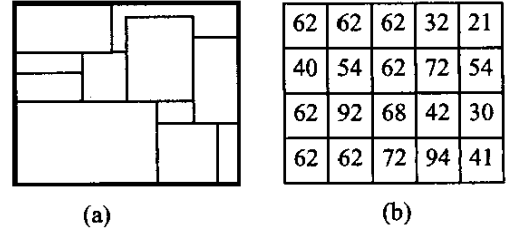| 62	62	62	32	21
40	54	62	72	54
62	92	68	42	30
62	62	72	94	41

(a)   (b)

Figure 4.1 (a) A floorplan, and (b) its tile graph .

Constraining the tiles to be lateral tiles, i.e., the length in x direction is not less than the length in y direction. The tile graph of power density automatically asserts the following two constraints:

1. $a \geq b > c$,
2. $x = a \times TD_x, y = b \times TD_y$.

Since the power dissipated by heat sources is evaluated only when these tiles are lateral tiles, the thickness of a circuit is far less than the length and width of a tile. In such case, Constraints (1) and (2) are automatically valid, this will simplify the eight cases of Fig. 2.1 into one case, i.e., Case 1, $t_{a1} \geq t_{b1} \geq t_{b2} \geq t_{a2} \geq t_c$. Instead of eight solutions for eight cases as in [5], the solution of Expression (1) under tile graph is exclusively expressed as follows.

$$\Delta T(x,y,0,\infty) = \frac{\alpha P\sqrt{t_c}}{4k(abc)}\left[(\sqrt{t_{b1}} - \sqrt{t_{b2}})(2.5 + \log\frac{t_{a1}}{t_{b1}}) + \right.$$

$$\left. (\sqrt{t_{a2}} + \sqrt{t_{b2}})\log\frac{t_{b2}}{t_{b1}}\right], \quad\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3a)$$

On the other hand, when the target tile is also a heat source tile, and assuming the tile size as $a \times b$, the self-heating solution of Expression (1) under tile graph can be expressed as follows.

$$\Delta T(0,0,0,\infty) = \frac{\alpha P}{4k(abc)}\left[-4t_c + 2\sqrt{t_b}\sqrt{t_c}(5 + 2\log\frac{t_a}{t_b})\right]\cdots (3b)$$

In Expression (3a), remember that the term t_c depends on the thickness of a heat source, while $t_{a1}, t_{b1}, t_{b2},$ and t_{a2} depend on the size (length and width) of a heat source as well as the distance from an observation point to a heat source. Defining the tile distance in x (y) component as TD_x (TD_y), the temperature on a target tile raised by a heat source tile can be derived as shown in Expression (4a).

$$\Delta T(TD_x, TD_y, 0, \infty) = \frac{P}{8\pi k}\left[5b + 2 \times \log(\frac{a + 2aTD_x}{b + 2bTD_y})^2 - (a + \right.$$

$$\left. b - 2aTD_x - 2bTD_y) \times \log(\frac{b - 2bTD_y}{b + 2bTD_y})^2\right] \quad\ldots\ldots\ldots\ldots(4a)$$

In the same manner, the temperature raised by self-heating can be expressed as shown in Expression (4b).

$$\Delta T(0,0,0,\infty) = \frac{P}{2\pi kab}(-2c + 5bc + 4bc\log\frac{a}{b})\ldots\ldots\ldots\ldots(4b)$$

As every item enclosed in brackets at the right side of Expression (4a) is composed of TD_x and TD_y. Assuming that every circuit blocks in a chip has equal thickness, i.e., c is a constant, the product of items enclosed in brackets and $1/8\pi k$ can be expressed as $F(TD_x, TD_y)$, which may be thought as a heat diffusion factor. Expression (4a) and (4b) can be further rewritten as Expression (5a).

$$\Delta T(TD_x, TD_y, 0, \infty) = P \times F(TD_x, TD_y),\ldots\ldots\ldots\ldots(5a)$$

Although Expression (5a) is directly applicable only to gray colored tiles in Fig. 4.2, the temperature raised on the tiles located at other region can be easily obtained from Expression (5b), a dual expression of Expression (5a), which is valid from the fact that the heat diffusion depends on the tile distance rather than on the direction.

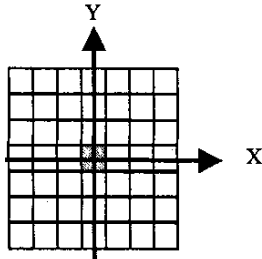$$\Delta T(TD_x, TD_y, 0, \infty) = P \times F(TD_y, TD_x)\ldots\ldots\ldots\ldots(5b)$$

Figure 4.2 Gray colored tiles reveal region that Expression (5a) directly applicable, $F(0, 0)$ of the dark-gray tile is derived from Expression (4b), $F(TD_x, TD_y)$ of other tiles in light-gray are derived from Expression (4a).

Now, we can build a look-up table (LUT) according to $F(TD_x, TD_y)$, as shown in Fig. 4.3(a), in which each element in LUT has two subscript indices which separately represent TD_x and TD_y, or, alternatively, TD_y and TD_x. For example, F_{13} stores $F(1, 3)$ or $F(3, 1)$.

Fig. 4.3(b) shows an example of deciding the temperature rise on a target tile caused by the power dissipation of a heat source tile, in which T_{15} is treated as a heat source tile and T_8 a target tile. As TD_x and TD_y are respectively 2 and 1, from F_{12}, we can immediately obtain the temperature raise on tile T_8 caused by the power dissipation of tile T_{15}, i.e., $\Delta T(2, 1, 0, \infty)=P_{15} \times F_{12}$.

For a tile graph of n tiles, although a LUT can relax the time pressure of computation, we still have to compute the temperature on each tile raised by n heat source tiles, and the complexity of $O(n^n)$ is unaffordable. To decrease the complexity, a normalization procedure is adopted for

selecting top 4 tiles dissipating distinguishable power density as candidates of hot tile. Take Fig 4.1(b) as an example, after normalization procedure, tiles with weight 94, 92, 72, and 72 will be selected as candidates of a hot tile, and the temperature raised on these four candidates will dominate the reliability of a floorplan as shown in Fig 4.1(a) in terms of heat.
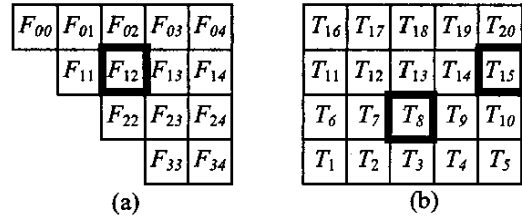

(a)   (b)

Figure 4.3 (a) LUT, (b) using F_{12} in LUT to efficiently decide the temperature rise of tile T_8 caused by tile T_{15} in a 4×5 tile graph.

Boolean Algorithm *Hot_Tile*

Input: candidates of hot tile $\{CT_0, CT_1, \ldots, CT_h\}$, a tile graph of power density $\{P_0, P_1, \ldots, P_n\}$, a LUT, and a threshold value ΔT_{th}.

1. for each candidates of hot tile CT_h
2. $\Delta T=\Delta T(0,0,0,\infty)$;
3.   for each tile of power density P_n
4.   find TD_x, TD_y between CT_h and P_n;
5.   take TD_x, TD_y as indices, find F from LUT;
6. $\Delta T = \Delta T + P_n \times F$;
7.   if $(\Delta T > \Delta T_{th})$ return true;
8. return false;

Figure 4.4 Algorithm for finding a hot tile.

To integrate the power planning into an iterative floorplanning algorithm such as simulated annealing (SA), the LUT is built before SA, normalization procedure and the algorithm as shown in Fig. 4.4 for finding any hot tile of a floorplan are taken after a new floorplan is generated by the perturbation function and before calculating the cost function of new generated floorplan. If the algorithm *Hot_Tile* returns true, the newly generated floorplan has to be abandoned and SA will take perturbation function again.

5. EXPERIMENTAL RESULTS

The procedure of building and normalizing tile graph of power density as well as algorithm *Hot_Tile* has been integrated into a floorplanner based on B*-tree [2]. The code is implemented in C++ on a Sun Sparc machine with 240MB memory. We tested the code on the MCNC benchmark circuits. The test results are summarized in Table 5.1, in which the power dissipation of each block is directly transformed from the current of each block. Besides, listed time and area are taken from the best of 20 runs.

Table 5.1 Test results on a B*-tree floorplanner.

Test circuits	tile-graph	without power planning		with power planning	
		time(s.)	dead space(%)	time(s.)	dead space(%)
ami33	6×6	6.1	4.58	8.6	6.88
apte	3×3	0.6	2.03	1.2	6.37
xerox	3×3	0.1	3.71	1.2	4.53

In Table 5.1, the tile-graph of power density is assumed to be 6×6 or 3×3 for accommodating the circuit size; a given floorplan is determined as failed if the accumulated ΔT of any candidate tile is larger than ΔT_{th}. As shown in [5], the factor of package is not considered and the temperature_rise in these experiments are relative values rather than absolute values. For explicitly showing the effectiveness of our approach, parameter k of Expression (4a) and (4b) in these experiments is deliberately set to a critical value such that hot tiles exist in some floorplans. The results reveal that the execution time is fast and the penalty of additional area is negligible.
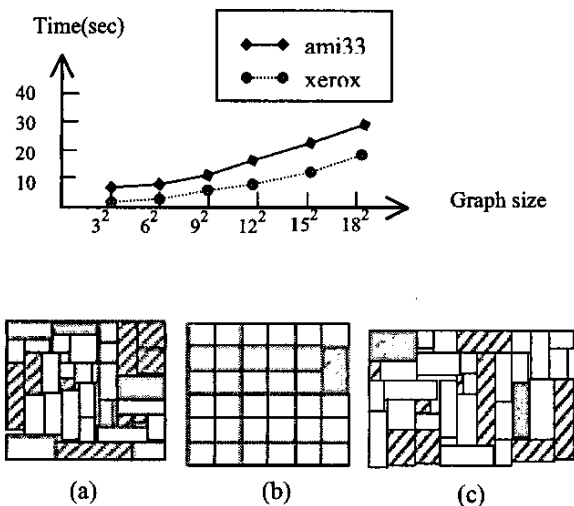

Fig. 5.1 reveals the influence of the tile graph sizes on the computation time, in which ami33 and xerox are taken as the testbenches. The result shows the pressure of time complexity $O(n^n)$ is relaxed by the normalization procedure.

On the other hand, Fig. 5.2 contrasts the power dissipation and temperature distribution of two floorplans of ami33. Fig. 5.2(a) shows a floorplan failed in Hot_Tile check; its associated hot tiles are shown in the tile graph of Fig. 5.2(b); and the floorplan that has passed hot-tile check is shown in Fig. 5.2(c). In Fig. 5.2, randomly assigned power of circuit blocks range from 2.1 mw to 9.0 mw, circuit blocks with 9.0mw power are marked solid red, circuit blocks with 7.6mw power are marked slashed

Figure 5.2 (a) A floorplan failed in Hot_Tile check, (b) reported tile graph of (a), and (c) a regenerated floorplan passed the Hot_Tile check.

red line, the others are empty.

6. CONCLUSION

The usefulness of a floorplanner depends not only on the ability of area packing, but also on the reliability of its solution. To provide an optimized solution and take care of the reliability of the solution as well has become a critical design issue.

The method proposed by [5] had proved its effectiveness. However, in a SOC, as the amount of cells integrated are enormous, the computation time of heat analysis will significantly lengthen each iteration of an iterative floorplanning algorithm.

We extend and strengthen the method of [5] such that a floorplanner can simultaneously estimate both the area and take care of the heat problem for each floorplan solution rather than identifying the hot spots of a final floorplan solution. The experimental results show the reasonable efficiency of our approach in practice and thus the effectiveness of its integrating into a floorplanning algorithm.

REFERENCES

[1] D. F, Wong, and C. L. Liu, "A new algorithm for floorplan design," Proc. DAC, pp. 101-107, 1986.

[2] Y.C. Chang, Y.-W. Chang, G.-M. Wu, and S.-W.Wu, "B*-tree: A new Representation for Non-Slicing Floorplans," Proc. DAC, pp. 458-463, 2000.

[3] I.-M. Liu, H.-M. Chen, T.-L. Chou, Adnan Aziz, D. F. Wong, "Integrated Power Supply Planning and Floorplanning," Proc. ASP-DAC, pp. , 2001.

[4] Shuo Zhou; Sheqin Dong; Xiaohai Wu; Xianlong Hong, "Integrated floorplanning and power supply planning," ASIC, Proc. 4th International Conference, pp.194-197, 2001.

[5] Yi-Kan Cheng, Sung-Mo Kang, "An Efficient Method for Hot-spot Identification in ULSI Circuits," IEEE Int. Conf. on Comput.-Aided Design, pp. 124-127, 1999.

[6] Amir H. Ajami, Kaustav Banerjee, Massoud Pedram, "Non-Uniform Chip-Temperature Dependent Signal Integrity," IEEE Symposium on VLSI Technology, pp. 145-146, 2001.