Area Reduction on Interconnect Optimized Floorplan using Deadspace Utilization

Chiu-Wing Sham, Evangeline F. Y. Young
{Department of Computer Science & Engineering, The Chinese University of Hong Kong, Hong Kong.
Tel: 852-26098344, fax: 852-26035302, e-mail:cwsham@cse.cuhk.edu.hk}

Abstract— Interconnect optimization has become the major concern in floorplanning. Many approaches would use simulated annealing (SA) with a cost function composed of a weighted sum of the area, wirelength and interconnect cost. These approaches can reduce the interconnect cost efficiently but the area penalty of the interconnect optimized floorplan is large. In this paper, we propose a new approach to reduce the area of interconnect optimized floorplan using deadspace utilization (DSU) by linear programming. The modules can be simple rectilinear shaped like L-shaped and T-shaped in floorplanning. Thus, some deadspace can be redistributed to the modules and the area of modules can be expanded. If the area of all the modules can be expanded by a same ratio using deadspace utilization, it means that the whole floorplan can be compacted by the same ratio. In addition, we can also limit the compaction ratio to prevent over-congestion. Experiments show that we can apply the deadspace utilization technique to reduce the area and wirelength of the interconnect optimized floorplan further and the constraint of the routability and congestion can be maintained at the same time.

I. Introduction

A. Motivations

Interconnect optimization has become the major concern in floorplanning. Due to the recent advances in VLSI technology, the number of transistors in a design are increasing rapidly, and interconnect has become a dominant factor in the overall performance of a circuit. Many floorplanning algorithms [2], [9], [10], [1], [5], [8], [4], [6] would use simulated annealing (SA) with a cost function composed of a weighted sum of the area, wirelength and interconnect cost. While the cost function is being minimized in the annealing process, the area, wirelength and routability cost are optimized accordingly. Although this can reduce the interconnect cost efficiently, the area penalty on the floorplan solution is not small.

Traditional floorplanning algorithms assumed that the modules are rectangular. Some floorplanners were further limited to mosaic or slicing floorplans for simplification. Actually, some simple rectilinear shapes such as L-shape or T-shape are acceptable because the modules are composed of standard cells which are rectangular shapes. In this way, the modules can be packed more closely together.

It will be very interesting if we can reduce the area of an interconnect optimized floorplan by making use of the flexibility of the modules while maintain the interconnect cost unchanged.

B. Related Works

Most existing floorplanning approaches only deal with rectangular modules. New approaches that can handle flexible and arbitrary shaped modules are essential to achieve high-performance design. However, floorplanning with flexible and arbitrarily shaped rectilinear modules is a complicated problem.

Both Kang et. at. and Xu et. at. [3], [11] propose that some integrated circuit components are not rectangular. Young et. at. [12] show that the area minimization problems with flexible modules can be solved optimally by geometric programming using Lagrangian Relaxation. However, the flexible modules are still rectangular in shape and the time penalty is large. Metha et. at. [7] presented three minimum-area floorplanning algorithms that assumes flexible and arbitrary rectilinear shapes. Those algorithms can minimize the area of the floorplan efficiently but modules of long-snake shaped can be resulted.

C. Our Contributions

In this paper, we assume an input floorplan that is first interconnect optimized. The deadspace in this input floorplan will be quite large. We will then apply a deadspace utilization technique to reduce the area and total wirelength of the floorplan. In floorplanning, the modules are not necessarily rectangular. Thus, some deadspace can be redistributed to the modules in order to reduce the total chip area. In addition, we would like to maintain the relative positions of the modules so that the wirelength of the floorplan can be reduced and the routability can be maintained.

We devised a linear program based method to perform deadspace utilization (DSU). By deadspace utilization, the area and wirelength of the interconnect optimized floorplan can be further reduced subject to the constraint of maintaining the routability and congestion of the original floorplan.

In this paper, we will first give an overview of our design in section II. The implementation details of deadspace utilization will be described in section III. The experimental results will be shown in IV.

II. Overview of Our Design

The modules in the floorplanning stage are not strictly rectangular. Deadspace utilization can be applied to reduce

the floorplan area subject to the constraint of maintaining the routability of the original interconnect optimized floorplan. An example is shown in figure 1.

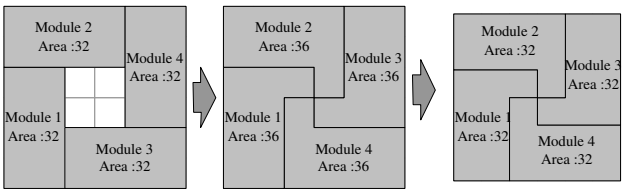

Fig. 1. An example of deadspace utilization

Given a floorplan, we will first find out all the deadspace blocks. Each deadspace block is surrounded by a number of modules. We will then assign room from the deadspace block to those modules in the surrounding to expand the occupying areas of the modules. If we can expand the occupying areas of all the modules by at least $\delta\%$, the whole floorplan can be compacted by $1 - \frac{1}{1+\delta\%}$. At the same time, the value of δ should be bounded by a congestion term because the net density of each tile will be increased as the area is reduced. Thus, δ should be bounded in order to maintain the routability of the floorplan. In the example of figure 1, modules 1, 2, 3 and 4 can be expanded by $\frac{4}{32}\% = 12.5\%$, $\frac{4}{32}\% = 12.5\%$, $\frac{4}{32}\% = 12.5\%$ and $\frac{4}{32}\% = 12.5\%$ respectively. It means that the whole floorplan can be compacted by $1 - \frac{1}{1.125} = 11\%$ altogether.

Notation	Description
m_k	Module k
M	The set of all modules
R_i	The set of deadspace blocks that are surrounded by module m_i
d_k	Deadspace block k
w_{d_k}	The width of deadspace block k
h_{d_k}	The height of deadspace block k
$A(d_k)$	The area of deadspace block k
D	The set of all deadspace blocks
S_i	The set of modules surrounds deadspace block d_i
$E_{i,j}$	The possible expanded area from the deadspace block d_j to module m_i where $m_i \in S_j$
$A(m_i)$	The area of module m_i
δ	The maximum area reduction ratio
G	The congestion constraint

TABLE I
NOTATIONS USED IN THIS PAPER

III. Implementation Details of Deadspace Utilization

A. Preprocessing of the Deadspace Blocks

When we apply deadspace utilization, we need to find all the deadspace blocks first. The deadspace blocks in the original floorplan may be of arbitrary rectilinear shapes. We will decompose the deadspace blocks into several smaller deadspace blocks which satisfy the following properties:

- Each side is surrounded by at most one module.
- Rectangular in shapes.

An example is shown in figure 2. After obtaining all these rectangular deadspace blocks, we can perform area minimization by linear programming.

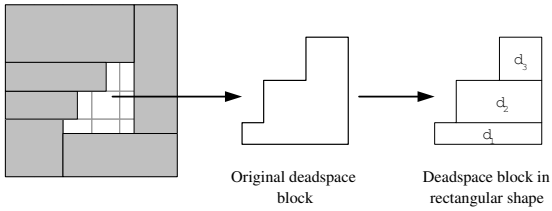

Fig. 2. Preprocessing of the deadspace blocks

B. Area Minimization by Linear Programming

In our design, we will try to find the maximum area reduction ratio $\delta\%$ by linear programming. We can at most use up the whole deadspace blocks, so we have the following set of equations:

$$\sum_{\forall m_i \in S_k} E_{i,k} \quad \leq A(d_k) \quad \forall d_k \in D$$

We should ensure that all the modules can be expanded by $\delta\%$, so we have the second set of equations.

$$\frac{\sum_{\forall d_i \in R_k} E_{k,i}}{A(m_k)} \quad \geq \delta\% \quad \forall m_k \in M$$

In addition, the value of δ should be bounded by a congestion constraint G (the floorplans are assumed to be interconnect optimized so we can obtain G according to the congestion estimations of the floorplans) such that when the whole floorplan is contracted by $1 - \frac{1}{1+\delta\%}$, the congestion at each tile is still below the maximum net capacity. Thus, the linear program is formulated as follows:

$$\begin{array}{lll} \text{Maximize} & \delta\% & \\ \text{Subject to} & \delta\% \leq G & \\ & \sum_{\forall m_i \in S_k} E_{i,k} \leq A(d_k) & \forall d_k \in D \\ & \sum_{\forall d_i \in R_k} E_{k,i} \geq A(m_k) \times \delta\% & \forall m_k \in M \end{array}$$

To solve the above linear programming problem, we will use the Simplex optimizer. The time complexity will depend on the number of equations. From table II, we can see that the average number of equations is linear to the number of modules in the floorplan. Thus, the above linear programming problem can be solved efficiently.

After we calculate the value of δ, we can change the dimensions of the modules proportionally to compact the floorplan, and the total wirelength will also be reduced accordingly.

Case	No. of modules	Average number of equations
hp	11	25.25
apte	9	18.75
*ami*33	33	54.75
*ami*49	49	80.75
playout	62	92.5

TABLE II
AVERAGE NUMBER OF EQUATIONS FOR EACH TEST CASES

C. Room Assignments of Deadspace Block

In order to assign room from a deadspace block to the modules in the surrounding, we will first divide the deadspace block into four sub-blocks and assign each sub-block to one module. At the beginning, the sub-blocks may be triangular or trapezoid in shapes. We require one more step to transform these sub-blocks into rectangular shapes. First, each deadspace block is surrounded by at most four modules and each side of the deadspace block can be surrounded by at most one module. We use m_L, m_R, m_T and m_B to denote the modules on the left, right, top and bottom of the deadspace block respectively. Then, the co-ordinates (x_1, y_1) and (x_2, y_2) with respect to the lower-left corner of d_j can be calculated according to table III.

When $(E_{L,j} + E_{R,j} > 0.5 \times A(d_j))$:
$x_1 = \quad w_{d_j} \times \frac{E_{L,j}}{E_{L,j}+E_{R,j}}$
$x_2 = \quad w_{d_j} \times \frac{E_{L,j}}{E_{L,j}+E_{R,j}}$
$y_1 = \quad E_{B,j} \times 2/w_{d_j}$
$y_2 = \quad h_{d_j} - E_{T,j} \times 2/w_{d_j}$
When $(E_{T,j} + E_{B,j} > 0.5 \times A(d_j))$:
$x_1 = \quad E_{L,j} \times 2/h_{d_j}$
$x_2 = \quad w_{d_j} - E_{R,j} \times 2/h_{d_j}$
$y_1 = \quad h_{d_j} \times \frac{E_{B,j}}{E_{B,j}+E_{T,j}}$
$y_2 = \quad h_{d_j} \times \frac{E_{B,j}}{E_{B,j}+E_{T,j}}$

TABLE III
CALCULATION OF (x_1, y_1) AND (x_2, y_2)

After we have computed the co-ordinates (x_1, y_1) and (x_2, y_2), we can divide the deadspace block into four sub-blocks. They are denoted by sb_R, sb_L, sb_T and sb_B. An example is shown in figure 3. We can see that each sub-block will abut with one module only and will later be assigned to that module. If $E_{L,j} + E_{R,j}$ is larger than half of the area of d_j, the sub-blocks will be divided as shown in figure 3a. If $E_{T,j} + E_{B,j}$ is larger than half of the area of d_j, the sub-blocks will be divided as shown in figure 3b.

In the second step, we will transform the sub-blocks into rectangular shapes. After the first step, four slanted lines will be formed. We can simply change the slanted lines to Z-shaped lines to transform the sub-blocks into rectangular shape. transformations. An example is shown in figure 4. We can see that we can obtain the rectilinear shaped sub-blocks by changing all the slanted lines to Z-shaped lines.

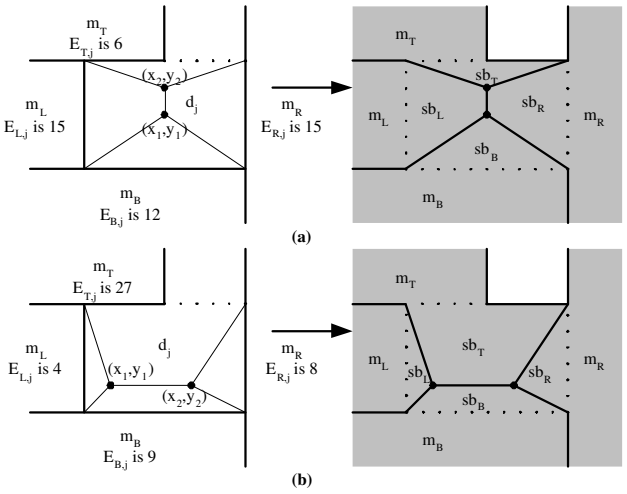

Fig. 3. Initial room assignments of deadspace block

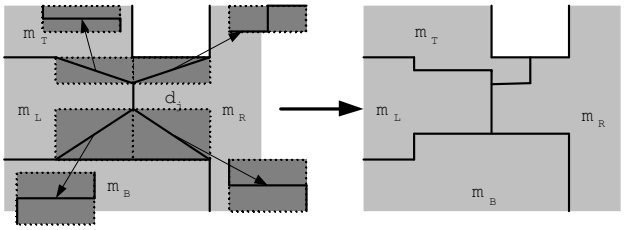

Fig. 4. Final room assignments of deadspace block

IV. EXPERIMENTAL RESULTS

In the experiments, we will show the improvement in area and wirelength by the deadspace utilization technique. We have implemented the linear programming method to perform deadspace utilization. All programs were written in C language and run on a machine with an Intel Xerox $2GHz$ processor and $1GMb$ memory.

We first use a floorplanner to obtain four different floorplans for each test case. The test cases used are MCNC benchmark circuits *hp*, *apte*, *ami*33, *ami*49 and *playout*. The detailed information of the testing circuits are shown in table IV. We applied the deadspace utilization technique to each floorplan. The value of the congestion constraint G is 20. It is calculated according to the net densities of the original floorplan. Finally, we will use a simple maze router to perform global routing and evaluate the routability of the floorplans. The multi-pin nets are decomposed into 2-pin nets by MST using center-to-center connections.

Cases	No. of cells	No. of nets	No. of 2-pin nets
hp	11	83	157
apte	9	97	183
*ami*33	33	123	305
*ami*49	49	408	526
playout	62	1611	2138

TABLE IV
INFORMATION OF THE TEST CASES

Cases	Area ($10^3 \mu m^2$)		Wirelength ($10^3 \mu m$)		No. of un-routable net		Runtime
	Before *DSU*	After *DSU*	Before *DSU*	After *DSU*	Before *DSU*	After *DSU*	(s)
hp	12.9[1.00]	12.1[0.93]	2.38[1.00]	2.30[0.96]	0.0	0.0	0.12
apte	54.0[1.00]	53.4[0.98]	6.58[1.00]	6.54[0.99]	0.0	0.0	0.15
*ami*33	626.3[1.00]	550.7[0.88]	22.08[1.00]	20.71[0.93]	0.0	0.0	0.18
*ami*49	71957.9[1.00]	64463.8[0.89]	495.60[1.00]	469.08[0.94]	0.0	0.0	0.19
playout	1798.1[1.00]	1623.5[0.90]	451.25[1.00]	428.79[0.95]	0.0	0.0	0.20

TABLE V
THE IMPROVEMENTS OF DEADSPACE UTILIZATION

In table V, the improvements in area, wirelength and routability brought by the deadspace utilization technique are shown. From the experimental results, we can see that the area and wirelength can both be reduced by the deadspace utilization step. As the area is reduced by the linear programming effectively, the distances between the modules are shorter. The wirelength can thus be reduced accordingly. The results also show that the improvements in area and wirelength are very significant (more than 10%) when the size of the floorplan is large. An example is shown in figure 5.

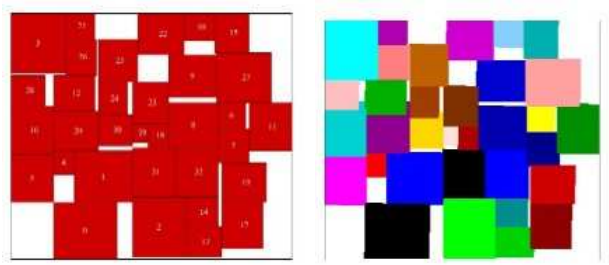

Fig. 5. Example of deadspace utilization

Although the shapes of some modules are changed, their shapes will not be very irregular and no snake-liked shapes will be resulted. In general, most of them are L-shaped and T-shaped only. It is because most modules will get new space from one deadspace block only according to the solution of the linear program. In addition, the routability of the floorplans can be maintained. It is because the reduction ratio is bounded by the congestion constraint G in the linear program.

V. Conclusion

To conclude, most existing interconnect-driven floorplanning approaches may lead to large penalty in chip area. In addition, the modules in floorplanning are not strictly rectangular in shape. We propose a new approach to reduce the area of an interconnect optimized floorplan using deadspace utilization by linear programming. Experiments show that we can apply the deadspace utilization technique to reduce the area and wirelength of the original floorplan further subject to the constraint of maintaining the routability and congestion of the floorplan.

References

[1] C. C. Chang, J. Cong, D. Z. Pan, and X. Yuan. Interconnect-driven floorplanning with fast global wiring planning and optimization. In *Proc. SRC Tech. Conference*, 2000.

[2] H. M. Chen, H. Zhou, F. Y. Young, D. Wong, H. H. Yang, and N. Sherwani. Integrated floorplanning and interconnect planning. In *Proceedings of IEEE Internation Cnference on Computer-Aided Design*, pages 354–357, 1999.

[3] M. Z. Kang and W. W. M. Dai. Arbitrary rectilinear block packing based on sequence pair. In *Proceedings of IEEE Internation Cnference on Computer-Aided Design*, pages 259–266, 1998.

[4] S. T. W. Lai, E. F. Y. Young, and C. C. N. Chu. A new and efficient congestion evaluation model in floorplanning: Wire density control with twin binary trees. In *Proceedings of Design, Automation and Test in Europe Conference and Exhibition*, 2003.

[5] J. Lou, S. Krishnamoorthy, and H. S. Sheng. Estimating routing congestion using probabilistic analysis. In *Proceedings of Internation Symposium on Physical Design*, pages 112–117, 2001.

[6] Y. C. Ma, X. L. Hong, S. Q. Dong, S. Chen, Y. C. Cai, C. K. Cheng, and J. Gu. Dynamic global buffer planning optimization based on detail block locating and congestion analysis. In *Proceedings of ACM/IEEE Design Automation Conference*, pages 806–811, 2003.

[7] D. P. Metha and N. Sherwani. On the use of flexible, rectilinear blocks to obtain minimum-area floorplans in mixed block and cell designs. In *ACM Transactions on Design Automation of Electronic Systems*, pages 82–97, 2000.

[8] C. W. Sham and E. F. Y. Young. Routability-driven floorplanning with buffer planning. In *IEEE Transactions on CAD of Integrated Circuit and System*, pages 470–480, April 2003.

[9] M. Wang and M. Sarrafzadeh. Modeling and minimization of routing congestion. In *Proceedings of the ASP-ACM/IEEE Design Automation Conference*, pages 185–190, 2000.

[10] M. Wang, X. Yang, and M. Sarrafzadeh. Congestion minimization during placement. In *IEEE Transactions on CAD of Integrated Circuit and System*, pages 1140–1148, October 2000.

[11] J. Xu, P. N. Guo, and C. K. Cheng. Sequence-pair approach for rectilinear module placement. In *IEEE Transactions on CAD of Integrated Circuit and System*, pages 484–493, April 1999.

[12] E. F. Y. Young, C. C. N. Chu, W. S. Luk, and Y. C. Wong. Floorplan area minimization using lagrangian relaxation. In *Proceedings of Internation Symposium on Physical Design*, pages 174–179, 2000.