Efficient Thermal Placement of Standard Cells in 3D ICs using a Force Directed Approach

Brent Goplen
Department of Electrical Engineering,
University of Minnesota, Minneapolis, MN 55455
bgoplen@ece.umn.edu

Sachin Sapatnekar
Department of Electrical Engineering,
University of Minnesota, Minneapolis, MN 55455
sachin@ece.umn.edu

Abstract

As the technology node progresses, thermal problems are becoming more prominent especially in the developing technology of three-dimensional (3D) integrated circuits. The thermal placement method presented in this paper uses an iterative force-directed approach in which thermal forces direct cells away from areas of high temperature. Finite element analysis (FEA) is used to calculate temperatures efficiently during each iteration. Benchmark circuits produce thermal placements with both lower temperatures and thermal gradients while wirelength is minimally affected.

1. Introduction

Three dimensional ICs are produced by stacking multiple active layers into a monolithic structure, using special processing techniques such as silicon-on-insulator (SOI) technology or wafer bonding techniques [2]. 3D ICs have several advantages over 2D ICs including higher transistor packing densities and shorter interconnect lengths, but thermal effects are expected to be more pronounced in 3D ICs [1]. With the advent of better processing technologies for 3D ICs, design tools are needed to realize their full potential and overcome thermal and efficiency issues. Current design tools used for 2D ICs can not be easily extended to 3D ICs [1] especially when taking into account thermal effects. In addition, efficient thermal placement algorithms are lacking even with 2D ICs.

Previous work done in both 3D placement and thermal placement has been quite limited. Tansprasert presented a 3D placement technique in [9] using an analytical model that reserved routing space and used a nonlinear programming technique. In [8], Reber and Tielert presented a 3-D placement tool for vertically stacked integrated circuits and used a simulated annealing algorithm to minimize a cost function containing terms for interconnect length, overlap, and testability. The 3D placement methods presented in both [8] and [9] fail to address thermal issues and lack run time efficiency.

Chu and Wong presented a matrix synthesis method for the thermal placement of gate arrays by evenly distributing sources of heat [4]. In [5], Eisenmann and Johannes suggested that their force-directed method could potential be used for distributing cells based on a heat map. Both of these approaches would lead to a uniform heat distribution, but Tsai and Kang pointed out in their paper [10] that a uniform heat distribution does not necessarily lead to a uniform temperature distribution.

Using the finite difference method (FDM) and simulated annealing, Tsai and Kang developed a thermal placement method for both standard cell and macro cell designs [10]. The thermal distribution was improved without sacrificing chip area or wire length, but it lacked efficiency. In [3], Chen and Sapatnekar presented a partitioning-based thermal placement method and improved upon the run time of the finite difference method presented in [10]. Despite their improvements, the method still appears to run in quadratic time if the number of thermal nodes is increased linearly with the circuit size.

The temperature computation methods presented in [3] and [10] lack the ability to simulate heat conduction between diagonally adjacent nodes in their fundamental formulation, requiring that additional internal nodes be added and inefficiently removed from the linear system of equations. The FEA method used in our paper uses only the nodes of interest with higher-order equations to accurately simulate lateral heat conductance between diagonally adjacent nodes. This reduces the number of nodes needed to produce accurate results, resulting in smaller systems of equations without sacrificing sparsity. Furthermore, the resulting linear system of equations is solved in nearly linear time.

2. Temperature Calculation

At steady state, heat conduction within the chip substrate can be described by the following differential equation [4]:

$$K\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2}\right)+Q(x,y,z)=0 \qquad (1)$$

where T is the temperature, K is the thermal conductivity, and Q is the heat generated per unit. A unique solution exists when convective, isothermal, and/or insulated boundary conditions are appropriately applied. The nature of the packaging and heat sink determines the boundary conditions.

2.1 FEA Background

In finite element analysis, the design space is first discretized or meshed into elements. An example of an 8-node hexahedral element is shown in Figure 1.

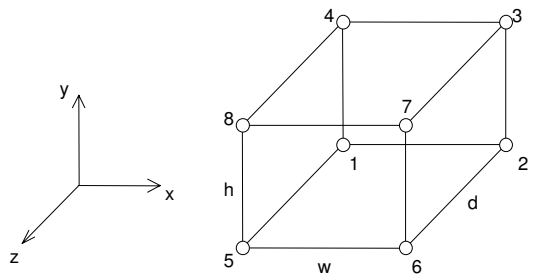

Figure 1. 8-node hexahedral element.

This work was supported in part by an SRC graduate fellowship and by the DARPA NeoCAD program.

ICCAD'03, November 11-13, 2003, San Jose, California, USA.

Proceedings of the International Conference on Computer Aided Design (ICCAD'03)
1092-3152/03 $ 17.00 © 2003 ACM

The temperatures are calculated at discrete points (the nodes of the element) and the temperatures elsewhere within the element are interpolated using a weighted average of the temperatures at the nodes. For an 8-node hexahedral element, the following interpolation function is used:

$$\mathrm{T}(\mathrm{x},\mathrm{y},\mathrm{z}) = [N]\{t\} = \sum_{i=1}^{8} N_i t_i \tag{2}$$

where $[N\ ]=[N_1\ \ N_2\ \ \ldots\ \ N_8\ ]$, $\{t\}=\{t_1\ \ t_2\ \ \ldots\ \ t_8\}^{\mathrm{T}}$, t_i is the temperature at node i, and N_i is the shape function for node i. The shape functions are determined by the coordinates of the element's center, (x_c,y_c,z_c), the coordinates at the nodes, (x_i,y_i,z_i), the width, w, height, h, and depth, d, of the element.

$$N_i = \left(\frac{1}{2} + \frac{2(x_i - x_c)}{w^2}(x - x_c)\right)\left(\frac{1}{2} + \frac{2(y_i - y_c)}{h^2}(y - y_c)\right)\left(\frac{1}{2} + \frac{2(z_i - z_c)}{d^2}(z - z_c)\right) \tag{3}$$

From the shape functions, the thermal gradient, $\{g\}$, can be found as follows:

$$\{\mathrm{g}\} = \left\{\frac{\partial T}{\partial x}\quad \frac{\partial T}{\partial y}\quad \frac{\partial T}{\partial z}\right\}^{\mathrm{T}} = [B]\{t\} \tag{4}$$

$$\text{where } [B] = \begin{bmatrix} \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial x} & \cdots & \frac{\partial N_8}{\partial x} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_2}{\partial y} & \cdots & \frac{\partial N_8}{\partial y} \\ \frac{\partial N_1}{\partial z} & \frac{\partial N_2}{\partial z} & \cdots & \frac{\partial N_8}{\partial z} \end{bmatrix}.$$

Similar to circuit simulation using the modified nodal formulation, stamps are made for each element and added to the global system of equations. In FEA, these stamps are called element stiffness matrices, $[k_c]$, and can be derived as follows using the variational method for an arbitrary element type [6]:

$$[k_c] = \iiint_V [B]^T [D][B]\,dV \tag{5}$$

$$\text{where } [D] = \begin{bmatrix} K & 0 & 0 \\ 0 & K & 0 \\ 0 & 0 & K \end{bmatrix} \text{ and } K \text{ is the thermal conductivity.}$$

2.2 Application of FEA

For a right prism with a width of w, a height of h, and a depth of d as shown in Figure 1, the element stiffness matrix is given in Equation (6) as an 8x8 symmetrical matrix with rows and columns corresponding to the nodes 1 through 8 [6]. For the entire mesh, the elements are aligned in a grid pattern with nodes being shared among at most 8 different elements. The element stiffness matrices are combined into a global stiffness matrix, $[K_{global}]$, by adding the components of the element matrices corresponding to the same node together. The global heat vector, $\{P\}$, contains power dissipated or heat generation as represented at the nodes. This is produced by distributing the heat generated by the cells among its closest nodes. A linear system of equations is produced, $[K_{global}]\{T\} = \{P\}$ with $\{T\}$ being a vector of all the nodal temperatures.

$$[k_c] = \frac{K}{36}\begin{bmatrix} +A & +B & +C & +D & +E & +F & +G & +H \\ +B & +A & +D & +C & +F & +E & +H & +G \\ +C & +D & +A & +B & +G & +H & +E & +F \\ +D & +C & +B & +A & +H & +G & +F & +E \\ +E & +F & +G & +H & +A & +B & +C & +D \\ +F & +E & +H & +G & +B & +A & +D & +C \\ +G & +H & +E & +F & +C & +D & +A & +B \\ +H & +G & +F & +E & +D & +C & +B & +A \end{bmatrix} \tag{6}$$

where $A = 4hd + 4wd + 4wh$, $B = -4hd + 2wd + 2wh$,
$C = -2hd - 2wd + wh$, $D = 2hd - 4wd + 2wh$,
$F = 2hd + 2wd - 4wh$, $E = -2hd + wd - 2wh$,
$G = -hd - wd - wh$, and $H = hd - 2wd - 2wh$.

3. Reducing the Global Matrices using Fixed Temperatures and Positions

Boundary conditions are applied to the global matrices using the following procedure [6], and this results in a reduced, nonsingular system of equations. Rows and columns that correspond to fixed nodal values within the global matrix are eliminated, as are corresponding values in the force or power vector (force and power are analogous), and the remaining values in the right-hand side vector are modified using the fixed values. For example, given the following system:

$$\left[\begin{array}{c:c} \mathrm{K}_{11} & \mathrm{K}_{12} \\ \hdashline \mathrm{K}_{21} & \mathrm{K}_{22} \end{array}\right]\left\{\begin{array}{c} \mathrm{X}_1 \\ \hdashline \mathrm{X}_2 \end{array}\right\} = \left\{\begin{array}{c} \mathrm{F}_1 \\ \hdashline \mathrm{F}_2 \end{array}\right\} \tag{7}$$

K_{11}, K_{12}, K_{21}, and K_{22} represent arrays of elements in the global stiffness matrix. X_1 are the unknown variables, and X_2 are fixed values. F_1 are known forces or power corresponding to the unknown values, X_1, and F_2 are unknown forces or power corresponding to the known values, X_2. This system can be reduced as follows:

$$[\mathrm{K}_{11}]\{\mathrm{X}_1\} = \{\mathrm{F}_1\} - [\mathrm{K}_{12}]\{\mathrm{X}_2\} \tag{8}$$

K_{11} is a nonsingular matrix, X_1 contain the unknowns, and the right-hand side is vector of constants so this linear system of equations can now be solved.

4. Force-Directed Placement Methods

Fundamentally, force-directed methodologies involve minimizing an objective function. Nets contribute certain costs to this objective function. The cost of a connection between nodes i and j is defined as [5]:

$$\mathrm{c}_{ij}\left((\mathrm{x}_i - \mathrm{x}_j)^2 + (\mathrm{y}_i - \mathrm{y}_j)^2 + (\mathrm{z}_i - \mathrm{z}_j)^2\right) \tag{9}$$

where c_{ij} is the weight of the connection between the two nodes. If the c_{ij} coefficients are combined into a global net stiffness matrix, [C], an objective function can be written for the entire system:

Proceedings of the International Conference on Computer Aided Design (ICCAD'03)
1092-3152/03 $ 17.00 © 2003 ACM

$$\frac{1}{2}\{x\}^T[C]\{x\}+\frac{1}{2}\{y\}^T[C]\{y\}+\frac{1}{2}\{z\}^T[C]\{z\} \quad (10)$$

where {x}, {y}, and {z} are the x, y, and z coordinates of all cells and points of interest. This objective function can be minimized by solving the following three systems of equations:

$$[C]\{x\}=\{f_x\},\ [C]\{y\}=\{f_y\},\ \text{and}\ [C]\{z\}=\{f_z\} \quad (11,12,13)$$

In the absence of external repulsive forces, the total force vectors, $\{f_x\}$, $\{f_y\}$, and $\{f_z\}$, would be zero. These three systems of equations are solved in the same way as in finite element analysis. Fixed coordinate values, created by physical constraints, can be used to reduce and solve the system of equations as shown in Equation (8). In an iterative force-directed approach, the forces, $\{f_x\}$, $\{f_y\}$, and $\{f_z\}$, are updated at each iteration [5].

4.1 Application of Force-Directed Method

Using a star model, a net is represented with a center node that connected to each cell and I/O pad in the net [7]. For each net, the following stamp is added to the global net stiffness matrix, [C] with connectivity, c_{ij}, assumed to be one:

$$\begin{array}{c|c|c|c} \vdots & +1 & -1 & \\ \text{cells in net} & \ddots & \vdots & \\ \vdots & +1 & -1 & \\ \hline \text{net center} & -1 \cdots -1 & +\text{cells}+\text{iopads} & -1 \cdots -1 \\ \hline \vdots & & -1 & +1 \\ \text{iopads in net} & & \vdots & \ddots \\ \vdots & & -1 & +1 \end{array} \quad (14)$$

A net representing the ground and Vdd supply is also added with its center constrained to the origin, thus ensuring that the placement stays centered around the origin.

Repulsive forces of zero are applied to the center node of each net so that it is positioned at the average of the net's cell and I/O pad locations, but non-zero repulsive forces are needed at the cell nodes in order to keep the entire system from collapsing into a single point. Forces of zero are applied to the I/O pads before using Equation (8). If there are *n* cells, *m* nets, and *l* I/O pads, then

$$\{x\}=\{x_{\text{cell}\,1}\ \cdots\ x_{\text{cell}\,n}\ \vdots\ x_{\text{net}\,1}\ \cdots\ x_{\text{net}\,m}\ \vdots\ x_{\text{iopad}\,1}\ \cdots\ x_{\text{iopad}\,l}\}^T,$$

$$\{y\}=\{y_{\text{cell}\,1}\ \cdots\ y_{\text{cell}\,n}\ \vdots\ y_{\text{net}\,1}\ \cdots\ y_{\text{net}\,m}\ \vdots\ y_{\text{iopad}\,1}\ \cdots\ y_{\text{iopad}\,l}\}^T,$$

$$\{z\}=\{z_{\text{cell}\,1}\ \cdots\ z_{\text{cell}\,n}\ \vdots\ x_{\text{net}\,1}\ \cdots\ x_{\text{net}\,m}\ \vdots\ x_{\text{iopad}\,1}\ \cdots\ x_{\text{iopad}\,l}\}^T,$$

$$\{f_x\}=\{F_{x_1}\ \cdots\ F_{x_n}\ \vdots\ 0 \cdots 0\ \vdots\ 0 \cdots 0\}^T,$$

$$\{f_y\}=\{F_{y_1}\ \cdots\ F_{y_n}\ \vdots\ 0 \cdots 0\ \vdots\ 0 \cdots 0\}^T,$$

and $$\{f_z\}=\{F_{z_1}\ \cdots\ F_{z_n}\ \vdots\ 0 \cdots 0\ \vdots\ 0 \cdots 0\}^T.$$

The thermal force, $\{f_t\}$, at a specific point in the placement area is equal to $-\{g\}$. The thermal gradient determines both the direction and magnitude of the thermal forces, thereby moving cells away from areas of high temperature.

5. Implementation

```
FORCE DIRECTED THERMAL PLACEMENT
{
        CONSTRUCT NET STIFFNESS MATRIX
        CONSTRUCT THERMAL STIFFNESS MATRIX
        RANDOM PLACEMENT
        CALCULATE INITIAL FORCES
        WHILE PLACEMENT IS IMPROVING
        {
                UPDATE POWER DISTRIBUTION
                CALCULATE TEMPERATURES
                DETERMINE THERMAL FORCES
                DETERMINE OVERLAP FORCES
                UPDATE REPULSIVE FORCES
                CALCULATE NEW POSITIONS
        }
        POST-PROCESS
}
```

Figure 2. Pseudocode of thermal placement method

The thermal placement algorithm is shown in Figure 2, and it begins by constructing the net and thermal stiffness matrices, placing standard cells randomly within the placement area, and multiplying the net stiffness matrix by the position vectors to get the initial forces. The placement procedure proceeds from here in an iterative fashion with the temperatures being calculated at the beginning of each iteration.

The repulsive forces, $\{f_{rep}\}$, are updated with thermal-dependent, $\{f_{therm}\}$, and overlap-dependent, $\{f_{ove}\}$, components using the following formula:

$$\{f_{rep}\}=k_{rep}\left((1-OP)\{f_{therm}\}+OP\{f_{ove}\}\right) \quad (15)$$

where k_{rep} is the repulsive force coefficient and *OP* is the percent of overlap force contribution to the repulsive force. $\{f_{therm}\}$ and $\{f_{ove}\}$ are normalized so that their average values are at the geometric mean of their raw values. The repulsive force coefficient is adjusted between iterations by multiplying the previous repulsive force coefficient by the ratio of the desired chip's dimensions to the previous chip's dimensions.

In order to assure continuity across element boundaries, $\{f_{therm}\}$ is calculated by averaging $\{f_t\}$ over adjacent elements. Overlap forces are determined using the procedure given in [5] and [7] with the exception that coarse groupings of elements are used for force contributions from far away areas. In order to prevent overshooting and instability in the iterative process, the new total force for a particular cell *i*, $\{F_{xi}, F_{yi}, F_{zi}\}^T$, is found by replacing only a small percentage, 5 to 10%, of the original total forces with the new repulsive forces. After the total forces are updated, new cell positions are calculated. At the end of the iteration, the convergence criteria are checked. If little improvements were made to the average cell temperature and wirelength within the last 10 iterations, the iterative placement process is stopped, and the placement is legalized using post processing.

In post processing, the cells are moved from continuous space to discrete space and made completely free of overlaps. First, the cells are sorted in the z direction, and placed into the nearest layer. If too many cells are placed in a particular layer, excess cells are placed in adjacent layers based on their initial proximity in the z direction. This process is repeated for the y direction in order to place cells in rows. Finally, a divide and

Proceedings of the International Conference on Computer Aided Design (ICCAD'03)
1092-3152/03 $ 17.00 © 2003 ACM

Table 1. Experimental results comparing force-directed placement with and without thermal forces.

Benchmark Circuit			Placement without Thermal Forces				Thermal Placement				
name	cells	nets	T_{ave}	T_{max}	g_{ave}	L_{total}	T_{ave}	T_{max}	g_{ave}	L_{total}	Time
biomed	6417	5743	95.6	174.2	89437	42.7	94.5	153.4	75031	43.8	70.7
ibm01	12282	11754	95.1	161.7	83375	63.8	93.0	130.1	68522	68.4	301.9
industry3	15059	21939	96.1	153.8	85918	119.8	95.0	141.7	79419	130.7	206.6
ibm03	22207	21905	95.8	148.0	83481	115.9	94.3	134.8	68646	132.3	665.0
ibm04	26633	26451	95.7	153.9	84360	144.5	94.8	136.7	73983	153.6	794.5
ibm06	32185	33521	96.1	153.6	83985	183.2	94.8	136.3	67197	189.2	1137.9
ibm07	45135	44682	96.5	144.7	82554	277.7	95.9	144.1	75202	274.4	1598.2
ibm08	50977	48231	97.0	151.8	83579	278.8	95.7	135.4	73184	281.5	1852.6
ibm09	51746	50679	96.3	145.1	83121	252.5	95.1	131.7	71473	283.4	1770.4

conquer approach is used to remove overlap within each row. Cells in a row are sorted in the x direction, and divided into two groups. As the two groups are divided, the overlap is removed between them. This continues recursively on each of the two groups until all overlap is removed.

6. Results

The program was written in C and run on an Intel Pentium 4 2.8 GHz machine with Linux. The conjugate gradient solver and ILU factorization preconditioner from the LASPack package [13] were used in our program to solve for the temperature and position values. The thermal placement method was tested using benchmark circuits from the MCNC suite [12] and the IBM-PLACE benchmarks [11] as shown in Table 1. Control results were generated with *OP* set to 100%, and the thermal placements had an *OP* of 50%. The substrate thickness was set to 700μm, the layer thickness was set to 10μm, and the interlayer thickness was set to 10μm. Four layers were used, and the chip size was fixed at 2cm x 2cm with the cell sizes adjusted accordingly.

The thermal conductivity of the silicon in the chip substrate was set to 150W/mC. The power dissipation of each cell was calculated by multiplying its area by a random value ranging from 0 to 2 x 10^7 W/m^2. In meshing the substrate, the number of elements used was increased linearly with the number of cells. The bottom and the sides of the chip were made isothermic with ambient temperature, and the top of the chip was made insulated in order to simulate the low heat sinking properties of the packaging substrate. The ambient temperature was set to 0°C for convenience, but the temperatures can be translated by the amount of any other ambient to reflect a different ambient temperature.

Overall, there was a 1.3% reduction in the average temperature, T_{ave}, a 12% reduction in the maximum temperature, T_{max}, and a 17% reduction in the average thermal gradient, g_{ave}, with a 5.5% increase in the total wirelength, L_{total}. With nearly linear time complexity per iteration and an almost constant number of iterations, the overall run time efficiency was also nearly linear.

7. Conclusion

An efficient three-dimensional thermal placement method was presented that attempts to overcome the thermal issues produced in the design of future ICs. The resulting placements have lower maximum and average temperatures with wirelength increasing slightly and the chip's average thermal gradient improving. The linear systems of equations produced by the force-directed placement and FEA are sparse and can be efficiently solved with conjugate gradient method in nearly linear time. The program also offers flexibility in applying positional constraints. It can allow a cell or I/O pad to be fixed in one, two, or all three dimensions. If all cells are fixed to a single plane, then thermal placement for 2D ICs can be performed. In the placement process, a continuous space placement is produced. Post processing eliminates any residual overlap and places cells into discrete layers and rows.

8. References

[1] K. Banerjee, S. J. Souri, P. Kapur, and K. C. Saraswat, "3-D ICs: A Novel Chip Design for Improving Deep Submicrometer Interconnect Performance and Systems-on-Chip Integration," in *Proc. of the IEEE*, vol. 89, no. 5, pp. 602--633, May 2001.

[2] M. Chan and P. K. Ko, "Development of a Viable 3D Integrated Circuit Technology," *Science in China*, vol. 44, no. 4, pp. 241-248, August 2001.

[3] G. Chen and S. S. Sapatnekar, "Partition-Driven Standard Cell Thermal Placement," in *Proc. of the ACM Int. Symp. on Physical Design*, pp. 75-80, 2003.

[4] C. N. Chu and D. F. Wong, "A Matrix Synthesis Approach to Thermal Placement," in *Proc. Int. Symp. on Physical Design*, pp. 163-168, 1997.

[5] H. Eisenmann and F. M. Johannes, "Generic Global Placement and Floorplanning," in *Proc. Design Automation Conf.*, pp. 269-274, June 1998.

[6] D. L. Logan, *A First Course in the Finite Element Method*, 3rd ed., Brooks/Cole Pub. Co., 2002.

[7] F. Mo, A. Tabbara, and R. K. Brayton, "A Force-Directed Macro-Cell Placer," in *Proc. of the Int. Conf. on Comput.-Aided Des.,* pp. 177-181, Nov. 2000.

[8] M. Reber and R. Tielert, "Benefits of Vertically Stacked Integrated Circuits for Sequential Logic," in *Proc. IEEE Int. Symp. on Circuits and Syst.*, vol. 4, pp. 121-124, May 1996.

[9] T. Tanprasert, "An Analytical 3-D Placement that Reserves Routing Space," in *Proc. IEEE Int. Symp. on Circuits and Syst.,* vol. 3, pp. 69 -72, 2000.

[10] C. H. Tsai and S. M. Kang, "Cell-Level Placement for Improving Substrate Thermal Distribution," *IEEE Trans. on Comput.-Aided Des.*, vol. 19, no. 2, pp. 253-266, Feb. 2000.

[11] http://er.cs.ucla.edu/benchmarks/ibm-place/

[12] www.cbl.ncsu.edu/pub/Benchmark_dirs/LayoutSynth92

[13] www.tu-dresden.de/mwism/skalicky/laspack/laspack.html

Proceedings of the International Conference on Computer Aided Design (ICCAD'03)
1092-3152/03 $ 17.00 © 2003 ACM