3D Module Placement for Congestion and Power Noise Reduction

Jacob R. Minz
School of ECE
Georgia Inst. of Technology
Atlanta, GA
jrminz@ece.gatech.edu

Sung Kyu Lim
School of ECE
Georgia Inst. of Technology
Atlanta, GA
limsk@ece.gatech.edu

Cheng-Kok Koh
School of ECE
Purdue University
West Lafayette, IN
chengkok@ecn.purdue.edu

ABSTRACT

3D packaging via System-On-Package (SOP) is a viable alternative to System-On-Chip (SOC) to meet the rigorous requirements of today's mixed signal system integration. In this work, we propose a 3D module and decap (decoupling capacitance) placement algorithm that simultaneously reduces the power supply noise and wire congestion. We provide efficient algorithms for 3D power supply noise and congestion analysis to guide our 3D module placement process. In addition, we allocate white spaces around the modules that require decaps to suppress the power supply noise while minimizing the area overhead. In our experimentation, we achieve improvements in both decap amount and congestion with only small increase in area, wirelength, and runtime.

Categories and Subject Descriptors

B.7.2 [**Design Aid**]: Placement and routing

General Terms

Algorithms, Design

Keywords

System-On-Package, 3D Module Placement, Congestion, Power Noise Reduction

1. INTRODUCTION

Semiconductor industry is beginning to question the viability of System-On-Chip (SOC) approach due to its low-yield and high-cost problem. Recently, 3D packaging via System-On-Package (SOP) [1] has been proposed as an alternative solution to meet the rigorous requirements of today's mixed signal system integration. The uniqueness of 3D SOP is in the highly integration of embedded RF, optical or digital functional blocks, and sensors, in contrast to stacked ICs and stacked package. Due to the high complexity in designing large-scale SOP under multiple objectives and constraints, computer-aided design (CAD) tools have become indispensable.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
GLSVLSI'05, April 17–19, 2005, Chicago, Illinois, USA.
Copyright 2005 ACM 1-59593-057-4/05/0004 ...$5.00.

Active devices in 3D packaging draw a large volume of instantaneous current during switching, which causes simultaneous switching noise (SSN). Moreover, wire congestions may create congested spots and thus increase the manufacturing cost of the 3D SOP due to the additional routing layers and vias needed. Existing approaches consider SSN and congestion issues as an afterthought, which may require excessive amount of decoupling capacitance (decap) and more routing layers. Thus, our goal in this work is to perform simultaneous SSN and congestion-aware physical design for 3D SOP.

In our approach, we first perform automatic placement of functional modules while minimizing the amount of decap required for each module. We then place decaps nearby the modules that need them while minimizing the overall footprint area. We developed a compact 3D SSN model that can be used to calculate the decap demand of a 3D solution efficiently. As a result, our method is much more efficient than directly employing SSN simulation during the optimization process. To the best of our knowledge, this is the first work to perform automatic placement of modules and decaps for 3D packaging. Our 3D wire congestion analysis consists of 3D pin redistribution and topology generation. Since 3D global routing update is time-consuming, we use trajectory-based method [2] to accurately estimate the congestion during Simulated Annealing process.

Decoupling capacitance placement is done for 2D PCB designs [3, 4] and 2D circuits [5, 6, 7]. Some of the recent congestion-driven 2D circuit placement works include [8, 9, 10]. Recently, physical design algorithms for 3D System-On-Package designs have been proposed [11, 12, 13, 14].

2. PROBLEM FORMULATION

Given the following as the input to our 3D SOP placement problem: (i) a set of blocks $B = \{B_1, B_2, \cdots, B_m\}$ that represent the various active and passive components in the given SOP design, (ii) width, height, and maximum switching currents for each block, (iii) a netlist $NL = \{n_1, n_2, \cdots, n_k\}$ that specifies how the blocks are connected via electrical wires, (iv) K, the number of placement layers in the 3D packaging structure, (v) the number of power/ground signal layers along with the location of the power/ground pins, and (vi) tolerance on simultaneous switching noise, minimize the following cost function:

$$w_1 \cdot A^{tot} + w_2 \cdot \sum_{n \in NL} wl_n + w_3 \cdot D^{tot} + w_4 \cdot C^{tot}$$

A^{tot} is the final footprint area. wl_n is the wirelength of net n. D^{tot} (section 3.3) is the total decap required and C^{tot} (section 3.2) is the wire congestion. In addition, decaps are required to be placed adjacent to the blocks that require them.

3. SOP PLACEMENT ALGORITHM

3.1 Overview of the Algorithm

Our decap/congestion-aware SOP placement algorithm is based on Simulated Annealing [15]. We extend the existing 2D Sequence Pair scheme [16] to represent our 3D module placement solutions. The algorithm starts with a random placement solution. The placement is perturbed at each move and the cost is calculated. The cost involves area, wirelength, congestion (section 3.2) and decap (section 3.3). The area is the final area after white space insertion for decap (section 3.4). The moves are accepted based on improvement over previous cost or some probability dependent on the difference. The moves are made until freezing point is reached. The final placement is compacted by Linear Programming (section 3.4).

3.2 3D Wire Congestion Estimation

Routing interval is defined as the interval between two adjacent placement layers. Let $G_i = (V_i, E_i)$ be the grid graph representing the routing resource for routing interval i. The routing density of an edge e in G_i, denoted d_e^i, is the total number of nets that use e. Then, $d^i = \max_{e \in E_i} \{d_e^i\}$ is the local congestion in routing interval i. The congestion of the 3D placement with K placement layers is given by $C^{tot} = \sum_i d^i$. Finally, the lower bound in the routing layers required for all routing intervals in a 3D package is $\sum_{i \in RI} d^i/cap$, where RI and cap respectively denote the set of all routing intervals and capacity of edges in G_i. We perform pin distribution, net distribution, and topology generation detailed in [17] to accurately measure C^{tot}.

Our congestion estimation involves several time-consuming steps and hence impractical to be used for each candidate solution. Trajectory method [2] for placement optimization consists of *sampling* solutions at appropriate points in the trajectory traced by the optimization routine and *collecting* the metrics (data-points) of the sampled solution. During congestion-aware placement, the data-points are used to estimate the congestion metric of all the candidate solutions. Let there be n collected points. The area, wirelength, and congestion at the ith point are denoted A_i, W_i, C_i, respectively. If A_c and W_c denote the area and wirelength for the *current* solution, the congestion C_c of the current solution is estimated using the following equations:

$$\begin{aligned}
d_i &= \|A_c - A_i\| + \|W_c - W_i\| \\
\bar{d} &= min\ d_i,\ i = 1, \cdots, n \\
\hat{w_i} &= e^{-\beta d_i/\bar{d}} \\
w_i &= \hat{w_i} / \sum_{i=1}^{n} \hat{w_i},\ i = 1, \cdots, n \\
C_c^i &= C_i + \left(\frac{\partial C}{\partial A}\right)_i (A_c - A_i) + \left(\frac{\partial C}{\partial W}\right)_i (W_c - W_i) \\
C_c &= \sum_{i=1}^{n} w_i \cdot C_c^i
\end{aligned}$$

The partial derivatives are calculated by taking the *previous* two values in the trajectory and calculating the gradient. The accuracy of estimated congestion values (C_c^i) depend on locality. The value of estimated congestion is to vary only within a certain range ($C_i - L < C_c^i < C_i + L$) to prevent pathological swings, where L is the limit. In our experiments β was fixed to 25. In the *adaptive* method, the trajectory is created during the optimization process. It has been shown [2] that the *adaptive method* is more efficient than the non-adaptive one, where the trajectory points are calculated beforehand. Thus, we usd this adaptive trajectory method in our placement.

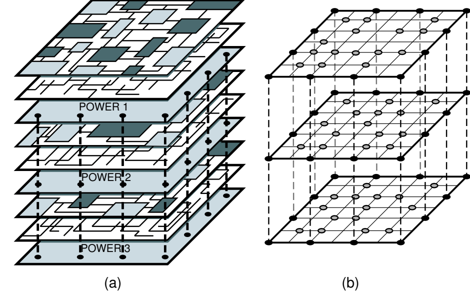

Figure 1: Illustration of 3D power supply modelling. (a) multi-layer power supply network, (b) 3D-grid modelling, where black and gray nodes respectively denote power supply and consumption nodes.

3.3 3D Power Supply Noise Modelling

We model the P/G network for 3D SOP as a 3D grid graph as shown in Figure 1. The edges in the grid-graph have inductive and resistive impedances. The mesh contains power-supply points and connection points, which supply and consume currents. The current distibution for the blocks is inversely proportional to the impedance of path from source.

The *dominant current source* for a block is defined as the voltage source supplying significantly more power to the block than any other neighboring sources. The *dominant path* for a block is the path from the dominant supply to the block causing the most drop in voltage. It has been shown experimentally in [5] that the shortest path between the dominant current source (nearest Vdd pins) and the block offers highly accurate SSN estimation within reasonable runtime. In our 3D SSN analysis engine, we compute dominant paths for all blocks which is dynamically updated whenever a new placement solution is evaluated in terms of SSN. Let P_k be a dominant current path for block k. Then $T^k = \{P_j : P_j \cap P_k \neq \emptyset\}$ denotes the set of dominating paths overlapping with P_k (T^k includes P_k itself). Let P_{jk} be the overlapping segments between path P_j and P_k. Let $R_{P_{jk}}$ and $L_{P_{jk}}$ denote the resistance and inductance of P_{jk}. After the current paths and their values have been determined for all blocks, the SSN for B_k is given by

$$V_{noise}^k = \sum_{P_j \in T^k} (i_j \cdot R_{P_{jk}} + L_{P_{jk}} \frac{di_j}{dt})$$

where i_j is the current in the path P_j, which is the sum of all currents through this path to various consumers. The weight of i_j and its rate of change are the resistive and inductive components of the path. Let Q^k denote the maximum charge drawn from the power supply by block B_k. If $\theta = \max(1, V_{noise}^k / V_{noise}^{lim})$, where V_{noise}^{lim} is the noise tolerance, the decap allocated to block B_k is given by

$$D^k = \frac{(1 - 1/\theta)Q^k}{V_{noise}^{lim}},\ 1 \leq k \leq M$$

where M denotes the total number of blocks to be placed. Finally, the decap cost is given by $D^{tot} = \sum_{k=1}^{M} D^k$.

3.4 3D Decoupling Capacitor Placement

In our LP-based 3D decap allocation shown in Figure 2, $x_k^{(j)}$ denotes the amount of decap allocated from white space k to block j. The objective is to maximize the utilization of white spaces for decap allocation. The constraint (2) limits the total allocation of a white space to its total area, where A_k denotes the area of white

LP-based 3D Decap Allocation

$$\text{Maximize} \quad S = \sum_{k=1}^{H} \sum_{j \in N_k} \beta_{kj} x_k^{(j)} \tag{1}$$

Subject to

$$\sum_{j \in N_k} x_k^{(j)} \leq A_k, \; k = 1, 2, \cdots, H \tag{2}$$

$$\sum_{k=1}^{k=H} x_k^{(j)} \leq S^{(j)}, \; j = 1, 2, \cdots, M \tag{3}$$

$$x_k^{(j)} \geq 0, \; \forall k, \forall j \tag{4}$$

Figure 2: LP-based 3D decap allocation

space k, and N_k denotes the neighboring blocks of k. The constraint (3) ensures that the total amount of decap allocated to each block does not exceed its demand, where $S^{(j)}$ denotes the demand determined through *SSN* analysis. β_{kj} evaluates the usefulness of whitespace k to be used as a decap for block j. The value of β_{kj} decreases with increasing distance of whitespace from the module that requires decap. Hence allocations of decap from far located whitespace will be avoided.

The white space (A_k) is generated by expanding the placement in the X and Y direction. The expansion is proportional to the decap demand of each module. In our Sequence Pair-based 3D placement, we modify the horizontal and vertical constraint graphs to expand the placement into X and Y directions, respectively. As exact decap allocation is a time-consuming process, it is impractical to solve an LP problem for every candidate solution. However, our white space insertion takes $O(n^2)$, which is computationally equivalent to computing block location using longest path computation for area. Therefore, we can afford white space insertion (and area increase) at every move. We perform the actual LP-based decap allocation at the end of the annealing process as a compaction stage.

4. EXPERIMENTAL RESULTS

We implemented our algorithms in C++/STL and ran our experiments on Linux Beowulf clusters. We tested our algorithms with two sets of benchmarks. The first set is from the standard GSRC circuits. The second set, named the GT benchmark, was synthesized from the IBM circuits [18], where we use our multi-level partitioner [17] to divide the gate-level netlist into multiple blocks. The GSRC benchmarks are small to medium-sized, while the GT benchmarks are medium to large, with *dense* connections. We chose area and wirelength-driven floorplanning, with *four* placement layer as our baseline. The technology parameters used for the SSN analysis are 0.01 $\Omega/\mu m$ for wire resistance per unit length, $1pH/\mu m$ for wire inductance per unit length, and $10fF/\mu m$ for wire capacitance per unit length. We summarize our observation from Table 1 as follows:

- Our *decap-aware* placer achieves a consistent decap saving (average of 20%) for all circuits at the cost of a small increase in final area (7%) and wirelength (15%).

- We used the trajectory optimization for the *congestion-aware* placement. The overall congestion improvement is 2% at the cost of 57% increase in area. However, the congestion is reduced as much as 12% in some cases. The runtime overhead related to the trajectory method is minimal (360 vs 401).

- Our *multi-objective* algorithm that considers all 4 objectives achieves a consistent improvement in both congestion and decap over the baseline with only a slight increase in area (13%) and wirelength (10%). An average improvement of 10% on decap and an reduction of congestion by 3% ,was achieved over the baseline.

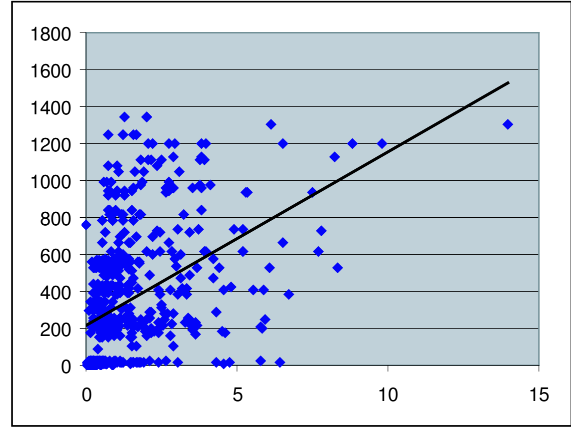

Figure 3: Decap vs congestion correlation plot for gt600. The correlation constant is 0.1934.

Table 2: Decap vs congestion correlation constants

ckt	correlation	ckt	correlation
n50	0.2188	n50b	0.0791
n50c	0.0013	n100	0.0662
n100b	0.0656	n100c	0.0048
n200	0.1433	n200b	0.1551
n200c	0.0909	n300	0.1786
gt100	0.2353	gt300	0.2422
gt400	0.1648	gt500	0.2028
gt600	0.1934	-	-

We report our correlation study between decap and congestion metric in Table 2. We notice that the correlation varies from low to medium, whereas most of the big circuit have moderate correlation. It is interesting to note that the trajectory estimation method works well with the circuits that show higher correlation. In this case, our simultaneous optimization of decap and congestion indeed helps reduce both metrics. Figure 3 shows a correlation plot for gt600, the biggest benchmark circuit.

Table 3 shows power supply noise simulation results for three 3D placement schemes: no-decap aware, decap-aware, and decap-aware+decap-placement. We report SSN noise for each block placed in the top placement layer. The P/G plane structure size is $246mm \times 254mm$, and the top P/G plane pair was modelled using cavity resonator model [19] and simulated in HSPICE. The placement layer that uses this P/G plan includes 14 active devices. The DC $5V$ sources are located at four edges in the plane pair and fourteen current sources exist in the plane pair. As can be seen from the Table 3, the SSN for the decap-aware algorithm is lower compared to non-decap-aware algorithm. The SSN of the noisiest block blk4 is $1.58V$, which is reduced to $1.42V$ by our decap-aware scheme. With the insertion of decap, the noise is suppressed to $0.22V$. In

Table 1: area/wirelength-driven vs decap-aware vs congestion-aware vs multi-objective (area, wirelength, decap, and congestion altogether) placement. The times are in seconds.

ckts	area/wire-driven				decap-aware				congestion-aware				multi-objective			
name	area	wire	decap	cong	area	wire	decap	cong	area	wire	decap	cong	area	wire	decap	cong
n50	22107	2.66	18.08	21	24674	2.99	9.15	22	40930	3.24	31.95	21	26217	3.02	18.26	20
n50b	25497	2.51	13.18	26	26263	2.76	4.1	22	34200	2.92	24.12	24	28383	2.67	6.88	24
n50c	21233	2.35	15.97	21	22940	2.92	2.23	23	26798	2.79	18.05	22	22169	2.84	3.15	26
n100	31551	6.66	78.23	41	34307	7.31	69.27	42	41482	7.60	85.49	39	37203	7.72	73.94	37
n100b	28014	4.83	73.04	35	30301	5.61	59.18	32	59916	6.56	108.23	35	33656	5.44	71.01	34
n100c	32886	6.21	80.68	41	34217	6.76	67.56	45	47210	6.98	105.59	40	35368	7.05	72.08	42
n200	56017	17.1	226.33	86	69395	20.5	223.22	74	81494	19.0	260.6	76	72780	19.9	224.29	79
n200b	55624	18.2	233.20	83	65354	21.3	224.34	79	100009	20.9	253.7	78	86011	22.8	237.31	67
n200c	54320	16.9	237.45	61	61356	19.6	228.6	76	123868	21.1	272.57	72	54315	16.9	237.45	66
n300	84620	28.6	393.87	101	84467	28.6	393.87	101	154232	35.0	418.04	93	84607	28.6	393.88	97
gt100	19156	132	60.85	1275	20743	168	42.55	1090	34690	172	88.13	1205	17509	132	54.79	1211
gt300	23898	196	342.52	1357	24858	223	334.99	1410	23630	196	341.86	1496	23642	198	338.24	1367
gt400	27017	281	493.14	1989	26851	325	482.05	1944	31858	285	512.56	1878	29873	294	505.18	1852
gt500	31665	303	645.35	1910	32169	354	632.41	2041	52813	398	685.8	1887	36653	322	664.61	1871
gt600	47541	608	806.50	2957	50903	711	794.8	2758	59078	626	828.3	2697	58405	673	821.09	2842
RATIO	1.00	1.00	1.00	1.00	1.07	1.15	0.80	0.99	1.57	1.17	1.24	0.98	1.13	1.10	0.90	0.97
TIME	360.00				392.54				401.15				432.41			

Table 3: SSN simulation results for area/wirelength-driven (no decap) and decap-aware placement. The actual decap amount for SSN supression is reported.

-	no	decap	decap	-	no	decap	decap
blk	decap	aware	used	blk	decap	aware	used
blk1	1.21	1.01	0.16	blk8	1.33	1.21	0.36
blk2	1.31	1.15	0.17	blk9	1.44	1.16	0.32
blk3	1.33	1.18	0.20	blk10	1.34	1.38	0.32
blk4	**1.58**	1.42	0.22	blk11	1.43	**1.44**	0.20
blk5	1.26	1.41	**0.50**	blk12	1.41	1.35	0.33
blk6	1.54	1.27	0.31	blk13	1.37	1.27	0.27
blk7	1.47	1.33	0.43	blk14	1.05	1.15	0.22

addition, the total amount of decap required for non-decap-aware algorithm is 26.7, which is reduced to 19.9 with our decap-aware scheme. The largest amount of decap is used for blk5 ($0.50nF$), because of an increase in its SSN after optimization. The numbers show that the SSN was efficiently suppressed and the amount of decap actually used is reduced by our decap-aware algorithms.

5. CONCLUSIONS

In this work, we proposed a 3D module and decap placement algorithm that simultaneously reduces the power supply noise and wire congestion. In addition, we allocated white spaces around the modules to suppress the power supply noise while minimizing the area overhead. We achieved improvements in both decap amount and congestion with only small increase in area, wirelength, and runtime.

6. REFERENCES

[1] R. Tummala, “SOP: What is it and why? a new microsystem-integration technology paradigm-moore’s law for system integration of miniaturized convergent systems of the next decade,” *IEEE Transactions on Advanced Packaging*, 2004.

[2] M. Rewienski and J. White, “A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices,” *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, pp. 155–170, 2003.

[3] A. Kamo, T. Watanabe, and H. Asai, “An optimization method for placement of decoupling capacitors on printed circuit board,” in *Proc. IEEE Electrical Performance of Electronic Packaging*, 2000.

[4] Y. Chen, Z. Chen, and J. Fang, “Optimum placement of decoupling capacitors on packages and printed circuit boards under the guidance of electromagnetic field simulation,” in *IEEE Electronic Components and Technology Conference*, 1996.

[5] S. Zhao, C.-K. Koh, and K. Roy, “Decoupling capacitance allocation and its application to power supply noise aware floorplanning,” *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, pp. 81–92, 2002.

[6] H. Su, S. Sapatnekar, and S. R. Nassif, “An algorithm for optimal decoupling capacitor sizing and placement for standard cell layouts,” in *Proc. Int. Symp. on Physical Design*, 2002, pp. 68–73.

[7] H. Chen, L. Huang, I. Liu, M. Lai, and D. Wong, “Floorplanning with power supply noise avoidance,” in *Proc. Asia and South Pacific Design Automation Conf.*, 2003.

[8] A. Ranjan, K. Bazargan, and M. Sarrafzadeh, “Fast hierarchical floorplanning with congestion and timing control,” in *Proc. IEEE Int. Conf. on Computer Design*, 2000.

[9] X. Yang, R. Kastner, and M. Sarrafzadeh, “Congestion estimation during top-down placement,” *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 2002.

[10] M. S. M. Wang, X. Yang, “Congestion minimization during placement,” *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 2000.

[11] R. Ravichandran, J. Minz, M. Pathak, S. Easwar, and S. K. Lim, “Physical layout automation for system-on-packages,” in *IEEE Electronic Components and Technology Conference*, 2004.

[12] J. Minz, S. K. Lim, J. Choi, and M. Swaminathan, “Module placement for power supply noise and wire congestion avoidance in 3D packaging,” in *Proc. IEEE Electrical Performance of Electronic Packaging*, 2004.

[13] J. Minz and S. K. Lim, “A global router for system-on-package targeting layer and crosstalk minimization,” in *Proc. IEEE Electrical Performance of Electronic Packaging*, 2004.

[14] J. Minz, E. Wong, and S. K. Lim, “Thermal and congestion-aware physical design for 3d system-on-pacakge,” in *IEEE Electronic Components and Technology Conference*, 2005.

[15] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, “Optimization by simulated annealing,” *Science*, pp. 671–680, 1983.

[16] 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.

[17] Jacob Minz, Sung Kyu Lim, ”A Global Router for System-on-Package Targeting Layer and Crosstalk Minimization,”in *IEEE Electrical Performance of Electronic Packaging*, 2004.

[18] C. J. Alpert, “The ISPD98 circuit benchmark suite,” in *Proc. Int. Symp. on Physical Design*, 1998, pp. 80–85.

[19] N. Na, J. Choi, M. Swaminathan, J. P. Libous, and D. P. O’Connor, “Modeling and simulation of core switching noise for asics,” *IEEE Trans. Advanced Packaging*, pp. 4–11, 2002.