

Automated Module Assignment in Stacked-Vdd Designs for High-Efficiency Power Delivery

Yong Zhan and Sachin S. Sapatnekar
 Department of Electrical and Computer Engineering
 University of Minnesota

Abstract—With aggressive reductions in feature sizes and the integration of multiple functionalities on the same die, bottlenecks due to I/O pin limitations have become a severe issue in today’s VLSI designs, especially for 3D IC technologies. To alleviate the pin limitation problem, a stacked-Vdd circuit paradigm has recently been proposed in the literature. However, for a circuit designed using this paradigm, a significant amount of power may be wasted if modules are not carefully assigned to different Vdd domains. In this paper, we present a partition-based algorithm for efficiently assigning modules at the floorplanning level, so as to reuse currents between Vdd domains and minimize the power wasted during the operation of the circuit. Experimental results on both 3D and 2D ICs show that compared with assigning modules to different Vdd domains using enumeration and simulated annealing, our algorithm can generate circuits with competitive power and IR noise performance, while being orders of magnitude faster.

I. MOTIVATION

3D IC technologies provide an effective vehicle for continuing device scaling along the Moore’s law trend by integrating multiple tiers of active devices in a single 3D chip [1] [2]. A prime attraction of this approach is that it provides a path for enhanced scaling that is orthogonal to the device scaling. However, this is certainly not the only advantage. 3D technologies can potentially reduce interconnect delays on critical wires significantly, reduce the overall congestion by permitting shorter routes, and facilitate the isolation of digital parts from analog/RF circuitry, thereby alleviating the substrate noise problem, and potentially allowing heterogeneous integration. For all of these reasons, there has been substantial research in this area in recent years.

On the design side, there are two major challenges associated with 3D ICs that must be overcome before the technology is ready for prime time. The first of these is related to the power that is dissipated within the 3D structure, which results in the generation of heat, and unless this heat is quickly transported away to the heat sink, it can lead to an increase in on-chip temperatures. This, in turn, can lead to leakage-temperature feedback cycles that further raise the temperature, and in the longer term, to reliability problems from devices and interconnects that undergo stress at these elevated temperatures. Fortunately, there has been extensive work in this area, including the development of methods for on-chip thermal analysis [3] [4] [5] as well as techniques for thermal optimization in 3D ICs, for example [6] [7] [8] [9] [10].

This work was supported in part by DARPA under the 3DIC program.

A second problem, which has seen substantially less research effort, is related to power delivery. Increasing the amount of on-chip circuitry leads to an increased demand in the power that must be delivered to the chip. Even for 2D ICs, this trend is evident from data in the International Technology Roadmap for Semiconductors (ITRS) [11]. Since the total number of I/O pins in a package is predicted to remain almost unchanged, the amount of current delivered per power pin tends to increase in the future as the performance of ICs improves. Fig. 1 plots data from the ITRS, showing the increasing trend of the current that must be delivered per power pin, as a function of the year. We refer to this trend as the I/O pin limitation problem, and this is the problem that is tackled in this paper.

The supply problem is particularly acute because the supply voltage value scales down with successive technology generations, implying that the tolerance for supply noise (in other words, the noise margin) becomes smaller with each generation. The higher current per I/O pin, going through the parasitics of the package and the chip, will lead to higher IR and $L\frac{di}{dt}$ noise, eating away at this reduced noise margin.

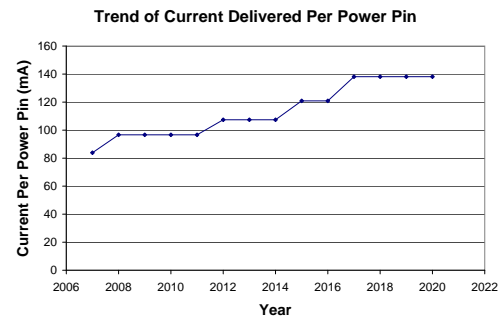


Fig. 1. Trend of current delivered per power pin based on the data from the ITRS [11].

For 3D ICs, the pin limitation problem has become much more severe due to the special structure of 3D chips. For implementing the same functionality, a 3D IC gains its advantage in achieving higher performance compared with its 2D counterpart by inte-

gration over the third dimension, which consequently reduces the footprint area of the chip and shortens the length of global interconnect wires. However, the reduced footprint area of a 3D IC has also reduced the number of I/O pins accessible to the circuit. In Fig. 2, we show the schematic of a transformation from a 2D IC to a k -tier 3D IC that implements the same functionality, where $k = 3$. It can be easily seen that because of the smaller footprint area, the number of I/O pins accessible to the 3D IC is only $1/k$ of that of the corresponding 2D IC. The net result is that this compounds the problem of scaling described in Fig. 1, and that each power pin must deliver an even larger amount of current to the 3D chip.

In principle, this is not surprising: by scaling technologies down and adding a third dimension, we compound the number of transistors on a chip. However, in practice, this leads to the very real and acute problem of developing techniques that can reliably deliver power to a 3D IC under the I/O pin limitation problem. In this paper, we develop an automated solution to this problem by employing the stacked-Vdd paradigm, to be described shortly. The solutions developed herein are applicable to both 2D and 3D ICs, but as seen above, the I/O pin limitation problem is much more critical for 3D chips, and for that reason, it is likely to benefit the 3D paradigm more significantly.

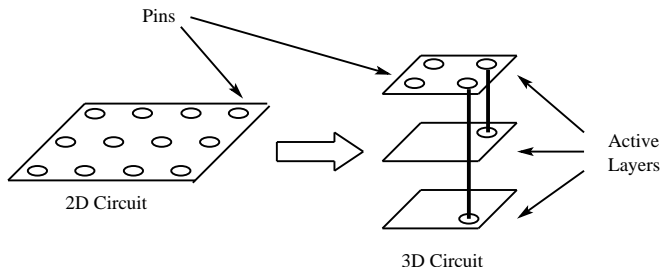


Fig. 2. The transformation of a 2D IC to a 3-tier 3D IC that implements the same functionality. The number of pins accessible to the 3D IC is reduced by $\frac{2}{3}$ compared with that of the 2D IC because of the much smaller footprint area.

II. DESIGN CONSIDERATIONS IN THE STACKED-VDD PARADIGM

The trend of increasing power consumption in high performance ICs is irreversible. However, the IR and $L\frac{di}{dt}$ noise in the power delivery network is not directly determined by power, but by the current delivered to the circuit. If, by some means, the current flowing through the power pins and on-chip power grids can be reduced, the signal integrity problem caused by the very limited number of power pins will see significant relief. In [12], a high-tension power delivery scheme was proposed to reduce power grid noise and the effect of electromigration. In this new circuit paradigm, logic blocks are stacked several levels high and power is delivered to the circuit as multiples of the regular supply voltage V_{dd} . Next, the delivered high-tension supply voltage is divided into several Vdd domains each of which has a range of V_{dd} , and circuit blocks are distributed to different Vdd domains.

Voltage regulators are used to control the voltage levels of internal supply rails.

An example of a two level stacked-Vdd circuit is shown in Fig. 3. The advantage of this new circuit structure is that when logic blocks are stacked n levels high and the current requirements between logic blocks operating in different Vdd domains are balanced, the current flowing through each external power grid would be reduced to $\frac{1}{n}$ of the original value, where the words external power grid refer to a power grid that is connected to power pins, i.e., nV_{dd} and GND rails in an n level stacked-Vdd circuit. Therefore, the noise and electromigration issues would be significantly alleviated.

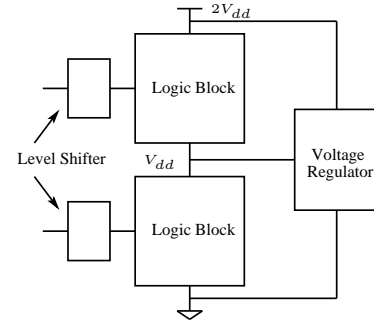


Fig. 3. A schematic of a 2-level stacked-Vdd circuit structure.

An important consideration in the design of a stacked-Vdd circuit is the current balance between logic blocks operating in different Vdd domains. If the currents are not balanced, the difference will flow through voltage regulators. This will not only lead to unnecessary power waste¹, but also increase the currents flowing through the external power grids, and therefore worsen the IR and $L\frac{di}{dt}$ noise. Fig. 4(a) shows an example of unbalanced current flow between modules operating in the two different Vdd domains. It can be seen that a current $|I_{2V_{dd}} - I_{V_{dd}}|$ will be wasted in the voltage regulator, where $I_{2V_{dd}}$ and $I_{V_{dd}}$ are the currents flowing through the two circuit blocks, respectively. The current balance issue has been addressed at the circuit level in [13] for architectures that contain parallel structures, where a data control circuitry is designed to distribute the work load at run time so as to reduce the current imbalance.

A more subtle issue associated with assigning circuit blocks to different Vdd domains is that the current balance must be maintained locally. The importance of this point can be clearly seen in Fig. 4(b). We assume that Block 1 and Block 2 are physically close to each other in the layout, and the same is true for Block 3 and Block 4. However, Block 1 and 2 are far away from Block 3 and 4. As a result, the resistance R associated with the internal power rail between the two parts of the circuit can not be ignored. Since the nodes marked N_1 and N_2 are both maintained at voltage level V_{dd} by regulators, there will be no

¹Because the power wasted in a voltage regulator is proportional to the current flowing through it given a constant supply voltage, we will use the words wasted power and wasted current interchangeably in this paper.

current flowing through the resistance R . If $I_1 = I_2$ and $I_3 = I_4$, local current balance is achieved and no power is wasted in the voltage regulators. If, on the other hand, $I_1 > I_2$, $I_3 < I_4$, but $I_1 + I_3 = I_2 + I_4$, then although the currents are still balanced globally, there will be a current $I_1 - I_2$ flowing through Regulator 1 and a current $I_4 - I_3$ flowing through Regulator 2 because there is no current flowing through the resistance R . Therefore, some amount of power will be wasted in the regulators under this situation.

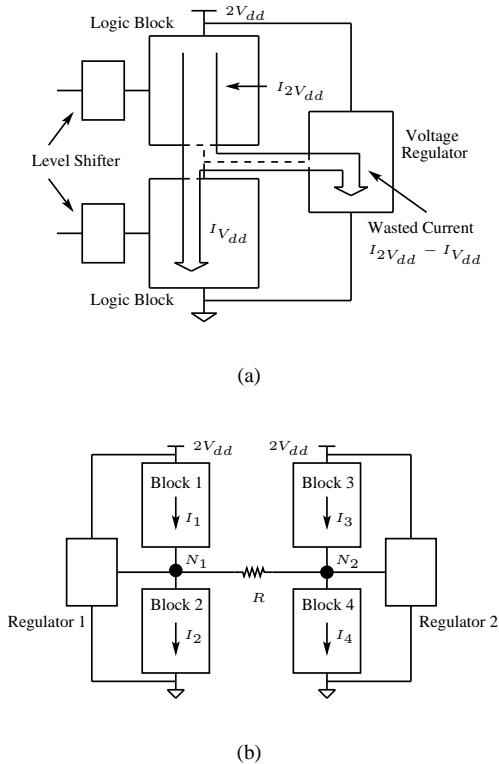


Fig. 4. Power wasted in voltage regulators (a) if the currents consumed by the logic blocks operating in the two different Vdd domains are not balanced, the difference will flow through voltage regulators and present itself as wasted power (b) current balance must be maintained locally in order to maximally reduce the power waste.

Another important issue that has to be considered in the design of a stacked-Vdd circuit is at which level should the circuit be partitioned into different Vdd domains. Note that a level shifter is required at the output of a logic block if it is used to drive another logic block operating in a different Vdd domain. If each logic block corresponds to a cell at placement level, too many level shifters will have to be used, which not only leads to a significant overhead in terms of silicon area, but also impairs the performance of the circuit because of the extra delays caused by level shifters. In this paper, we address the module assignment problem at the floorplanning level where the number of modules is usually not very large. Therefore, the performance degradation and the area waste caused by level shifters can be largely ignored.

In addition, unlike [13], we do not impose the restriction that the circuit contains parallel processing units. Instead, we utilize the observation that the operations of many modules on a VLSI chip are correlated, e.g., the modules on a pipelined data path tend to be on at the same time.

III. PROBLEM FORMULATION AND MODULE ASSIGNMENT ALGORITHM

There are several ways of trying to map the stacked-Vdd paradigm to 3D ICs. The simplest of these may be to allocate a single Vdd value to each tier of the 3D IC, and in case of a k -tier IC, for example, one could use a k -stacked Vdd paradigm. The problem with assigning a single Vdd to each tier is related to the current recycling constraints, described in the previous section, which dictate that blocks with similar currents must lie physically close to each other. This implies that high-current blocks will be stacked up over each other, which is highly suboptimal from the point of view of thermal issues. Hence, we do not use this approach; instead, given a k -stacked Vdd paradigm, we divide the circuitry in each 3D tier into segments that are allocated to each of the k stacks.

Because the two level stacked-Vdd circuit provides a good tradeoff between chip performance and engineering complexity, it will be the focus of this work, and the primary steps of a module assignment algorithm for 3D ICs could include:

- We begin with a floorplan for each tier that contains both regular modules and voltage regulators.
- For each tier, we use the power simulation results over a set of benchmark programs to characterize the correlation between modules, which is represented in the form of a graph.
- Using an iterative approach, we perform a max-cut partitioning of each graph obtained in the previous step, which corresponds to the assignment of modules to the two different Vdd domains.

As stated earlier, although the design of 3D ICs is the major motivation behind this work and we will refer to 3D ICs frequently in the paper, the presented algorithm applies equally well to 2D ICs since it corresponds to the degenerated case where the 3D IC has only one tier. The performance of the algorithm will be characterized using both 2D and 3D ICs in the experimental results section. In what follows, we will provide the details of the flow shown above, and special emphasis is placed on the partition-based module assignment step.

A. Problem formulation

In this section, we focus on a particular tier of a 3D IC and assume that a 2D floorplan for the tier that include both regular modules and voltage regulators is given. The primary objective is to assign modules to different Vdd domains so as to achieve the maximal current balance. The module assignment problem for a two level stacked-Vdd circuit is formulated as follows:

Given a floorplan including the location and size of each module and voltage regulator, the structure of the power grids, a set of current consumption traces of modules obtained through simulations over a set of benchmark programs, find the assignment of modules to the two different Vdd domains so that the total power wasted by voltage regulators is minimized.

Since the floorplan and the structures of the power grids are given as the input to the problem, we could run detailed simulations to obtain the trace of current flowing through each of the voltage regulators, and therefore calculate the wasted power. However, one simulation is required for each candidate assignment of modules, and the overall runtime of this scheme becomes exponential with respect to the number of modules in the layout, which makes it impractical for real design problems. In what follows, we will show how to estimate the current flowing through each voltage regulator and how to obtain the assignment of modules by solving a single graph partitioning problem.

B. Estimation of the current flowing through a regulator and the formulation of the graph partitioning problem

Note that the tapping points of voltage regulators to the internal power grid, i.e., the connecting points between voltage regulators and the V_{dd} rail, have properties that are similar to those of power pins. For well designed regulators, they provide rather stable voltage levels at V_{dd} . In [14], it was demonstrated that each module primarily draws currents from nearby power pins, and the same observation can be applied to the tapping points of voltage regulators. Assume we have K voltage regulators distributed across the tier. Each regulator is represented by the point it taps into the V_{dd} grid. As shown in Fig. 5, we can divide the tier into K regions accordingly such that there is one regulator in each region and the i^{th} region contains all the points on the tier that primarily draw (sink) currents from (to) the i^{th} regulator. The division of the tier into non-overlapping regions can be achieved through meshing the entire tier area using a fine grid and calculating the value of certain metric associated with each grid cell to determine which region it belongs to. This metric could be distance based, i.e., each cell belongs to the region that is controlled by the nearest voltage regulator, or it could be based on other criterion determined by our understanding of the power grids and the accuracy requirements. In our implementation, we choose to use the Euclidean distance as the metric, and therefore, the resulting division of the tier corresponds to the Voronoi diagram of the region enclosed by the tier boundary. However, we emphasize that our algorithm is not tied to the Voronoi diagram because the metric calculating part is an independent function and it can be easily modified to use a different metric.

After the tier is partitioned into disjoint regions, we can assume that the imbalance current caused by the modules located in a particular region only goes through the regulator in the same region. For example, if modules M_1 and M_2 are the only modules located in the region corresponding to tapping point R_2 , M_1 works between the $2V_{dd}$ and V_{dd} rails and draws a current I_1 ,

M_2 works between the V_{dd} and GND rails and draws a current I_2 , and $I_1 > I_2$, then a current $I_1 - I_2$ will flow through the voltage regulator tapped at point R_2 . If a module is located at the boundary between multiple regions, e.g., M_3 in Fig. 5, it will be decomposed into several submodules with one submodule in each region it overlaps and with the constraint that all submodules must be assigned to the same Vdd domain.

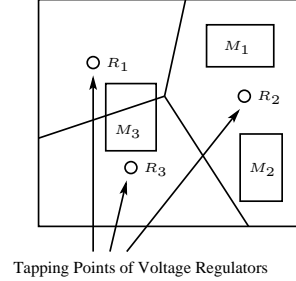


Fig. 5. Partitioning of the tier into disjoint regions each of which is controlled by a voltage regulator.

Let us focus on a particular region corresponding to a particular voltage regulator. Assume the modules located in this region are M_1, M_2, \dots, M_n , where the current flowing through module M_i as a function of time t is given by $I_i(t)$. Because voltage regulators can only respond to the low to mid frequency components of the imbalance currents while the high frequency components are usually handled by on-chip decaps, we pre-process the input current traces obtained through cycle-accurate power simulations to smooth out the high frequency components in the current signals. The smoothing process is performed by first dividing the entire time sequence of the simulated program into consecutive segments of clock cycles as shown in Fig. 6, and then for each segment, taking the average current consumption of each module. Therefore, $I_i(t)$ should be understood as containing only the low to mid frequency components of the current flowing through module M_i .

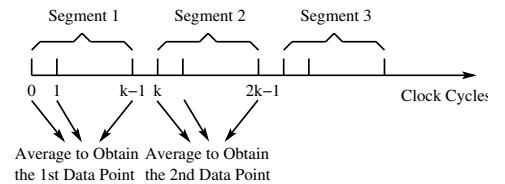


Fig. 6. Smoothing the current trace to remove the high frequency components.

If we associate a 0/1 integer variable x_i with module M_i defined as

$$x_i = \begin{cases} 0 & \text{if } M_i \text{ operates between the } 2V_{dd} \text{ and } V_{dd} \text{ rails} \\ 1 & \text{if } M_i \text{ operates between the } V_{dd} \text{ and } GND \text{ rails} \end{cases} \quad (1)$$

the total current flowing through the voltage regulator at time t , which is proportional to the instantaneous power wasted in the

same regulator, will be approximated by

$$\begin{aligned} I_R(t) &= \left| \sum_{i=1}^n I_i(t) \cdot (1 - x_i) - \sum_{i=1}^n I_i(t) \cdot x_i \right| \\ &= \left| \sum_{i=1}^n I_i(t) \cdot (1 - 2x_i) \right| \end{aligned} \quad (2)$$

The objective is to minimize the average wasted current $\overline{I_R(t)}$. It is shown in the appendix that this is a NP-hard problem that can, realistically, only be solved using heuristics. In our work, instead of minimizing $\overline{I_R(t)}$, we minimize $\overline{I_R(t)^2}$, which can be written as

$$\begin{aligned} \overline{I_R(t)^2} &= \overline{\left(\sum_{i=1}^n I_i(t) \cdot (1 - 2x_i) \right)^2} \\ &= \overline{\left(\sum_{i=1}^n I_i(t) \right)^2} - 4 \sum_{i < j} \overline{I_i(t) I_j(t)} (x_i + x_j - 2x_i x_j) \end{aligned} \quad (3)$$

When going from the first to the second step in equation (3), we used the equality $x_i^2 = x_i$ when x_i is a 0/1 integer variable. It is easy to see from (3) that to minimize $\overline{I_R(t)^2}$ is equivalent to maximize

$$S = \sum_{i < j} \overline{I_i(t) I_j(t)} (x_i + x_j - 2x_i x_j) \quad (4)$$

and

$$x_i + x_j - 2x_i x_j = \begin{cases} 0 & \text{if } x_i = x_j \\ 1 & \text{if } x_i \neq x_j \end{cases} \quad (5)$$

The intuition behind (4) and (5) is that if modules M_i and M_j are in different Vdd domains, a positive term $\overline{I_i(t) I_j(t)}$ will appear in summation (4), but not otherwise. Based on this observation, the problem of maximizing S in (4) can be cast into the following equivalent graph partitioning problem.

Given a weighted graph $G = (V, E, W)$ where $V = \{V_1, V_2, \dots, V_n\}$, $E = \{(V_i, V_j) | V_i, V_j \in V\}$, and the weight set $W = \{w(V_i, V_j) = \overline{I_i(t) I_j(t)} | V_i, V_j \in V\}$, find a two way partitioning of the graph so that the total cut of the edges crossing the partition is maximized.

Up to this point, we have been considering one of the K disjoint regions over the tier and the modules that are completely located within it. A graph partitioning problem is formulated to assign modules to the two different Vdd domains so as to minimize the power wasted in the voltage regulator controlling the region. For the entire tier, a similar graph partitioning problem can be formulated where node V_i in the graph corresponds to module M_i in the layout. The only difference from the problem formulation shown above is in calculating the weight of each edge in the graph. Let S_i represent the area of the i^{th} module, and denote the overlap area between the i^{th} modules and the k^{th} region over the tier by S_{ik} . The weight of edge (V_i, V_j) is

calculated by

$$w(V_i, V_j) = \left(\sum_{k=1}^K \frac{S_{ik} S_{jk}}{S_i S_j} \right) \overline{I_i(t) I_j(t)} \quad (6)$$

The intuition behind (6) is that for any pair of modules, only the portions that are located in the same region over the tier count toward the calculation of the correlation between them. A further implication of (6) is that if modules M_i and M_j are completely separated into two disjoint regions, the weight $w(V_i, V_j)$ will be zero, and therefore, the corresponding edge can be removed from the graph. In Fig. 7, we show the resulting graph corresponding to a tier that contains five modules and is divided into two regions. The circles marked R_1 and R_2 represent the tapping points of voltage regulators to the V_{dd} rail. Note that there is no edge connecting nodes V_2 and V_5 because modules M_2 and M_5 are completely separated into two disjoint regions controlled by two individual voltage regulators. Similarly, there is no edge between nodes V_3 and V_5 . For the modules that overlap with the boundary between the two regions, i.e., M_1 and M_4 , we assume that α portion of M_1 and β portion of M_2 are located in the region controlled by voltage regulator R_1 , and correspondingly, $1 - \alpha$ portion of module M_1 and $1 - \beta$ portion of module M_2 are located in the region controlled by voltage regulator R_2 . According to (6), the weights of edges (V_1, V_2) and (V_1, V_4) are calculated by

$$w(V_1, V_2) = \alpha \overline{I_1(t) I_2(t)} \quad (7)$$

and

$$w(V_1, V_4) = [\alpha\beta + (1 - \alpha)(1 - \beta)] \overline{I_1(t) I_2(t)} \quad (8)$$

respectively.

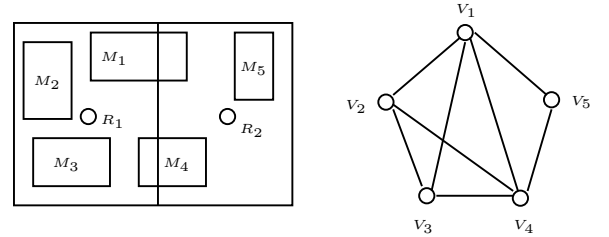


Fig. 7. An example of graph construction. Module M_i in the layout corresponds to node V_i in the graph.

C. Graph partitioning heuristic

A two step heuristic is used to partition the node set of the graph into two subsets so that the total cut is maximized. In the first step, we greedily assign nodes to the subsets so as to obtain a reasonably good initial partition. The primary operations involved in this step include sorting the weighted edges in decreasing weight order and examining them consecutively. For each edge under examination, if none of the two nodes associated with it has been assigned to a partition, we assign them to two different partitions. If one of the nodes has been assigned but not the other, we assign the other node to the opposite partition. Finally, if both

nodes have been assigned, we skip the current edge and proceed to the next edge in the sorted edge list.

After the initial node assignment, we use a F-M like algorithm to iteratively improve the partition and increase the cut size. Since the F-M algorithm is a rather mature method, we will not go into the details of every step of our implementation. Instead, we will highlight the differences between our algorithm and the conventional F-M algorithm, and then list our implementation in the form of a pseudo-code. Readers who are interested in the conventional F-M algorithm are referred to [15]. The first difference between our algorithm and the conventional F-M algorithm is that the latter requires the node weights to be balanced between the two partitions, while in our case, nodes carry no weight and maximizing the total cut size is our only concern. The second difference between the two algorithms is that while the F-M algorithm tries to minimize the cut size, our algorithm tries to maximize it. Therefore the calculation of the gain of each move should be modified.

The initial gain of moving a node V_i from its current partition to the opposite partition is given by

$$g(V_i) = \sum_{V_j \in FP(V_i)} w(V_i, V_j) - \sum_{V_j \in TP(V_i)} w(V_i, V_j) \quad (9)$$

where $FP(V_i)$ contains all the nodes that are in the same partition as node V_i and are connected to V_i , and $TP(V_i)$ contains all the nodes that are in the opposite partition to node V_i and are connected to V_i . For example, in the partition shown in Fig. 8, the initial gain of moving node V_2 from its current partition to the opposite partition is given by $g(V_2) = w(V_1, V_2) - w(V_2, V_4) - w(V_2, V_5)$.

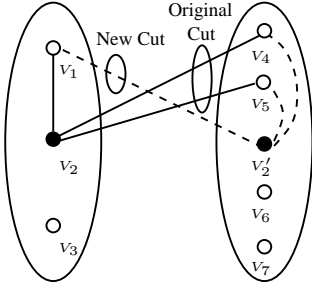


Fig. 8. An example for gain calculation in the F-M like algorithm showing the cut set before and after the node V_2 is moved to the opposite partition.

When a node V_i is moved from one partition to the other, the gains associated with the nodes connected to V_i must be updated. Assume node V_j is connected to V_i , the gain associated with V_j should be updated as follows:

- If V_j and V_i were in the same partition before the movement of V_i

$$g(V_j)^{new} = g(V_j)^{old} - 2w(V_i, V_j) \quad (10)$$

- if V_j and V_i were in different partitions before the movement of V_i

$$g(V_j)^{new} = g(V_j)^{old} + 2w(V_i, V_j) \quad (11)$$

The complete F-M like algorithm that is used to iteratively improve the cut size of the partition is listed in Fig. 9.

```

1) do
2)   node_moved ← false;
3)   Calculate the initial gain of each node;
4)   for i ← 1 to number_of_nodes
5)     Select the free node that has the maximum gain
       and call it  $V_{s(i)}$ ;
6)     Lock node  $V_{s(i)}$ ;
7)     Update the gains of the nodes connected to  $V_{s(i)}$ ;
8)   end for;
9)   Find the number  $K$  such that  $G = \sum_{i=1}^K g(V_{s(i)})$ 
       is maximized;
10)  if  $G > 0$ 
11)    Make the  $K$  moves permanent;
12)    node_moved ← true;
13)    Free all locked nodes;
14)  end if;
15) while( node_moved == true );

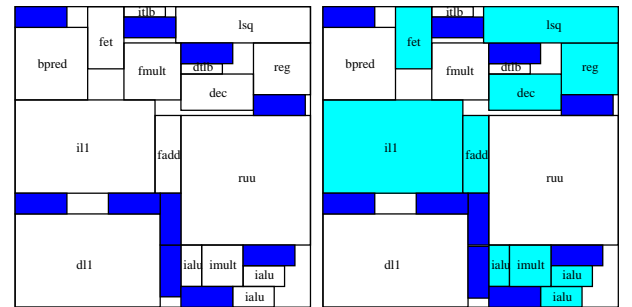
```

Fig. 9. Iterative improvement of the partition through a F-M like algorithm.

IV. EXPERIMENTAL RESULTS

We first show the results of module assignment for a 2D chip. This is not only because our algorithm is equally effective in both 2D and 3D designs, but also because the handling of each individual tier in a 3D design also uses these 2D routines. After the effectiveness of the algorithm on 2D designs are demonstrated, we will present the experimental results on 3D designs.

A. Calculation of the edge weight in the graph and module assignment using the partition-based algorithm



(a)

(b)

Fig. 10. Floorplan with voltage regulators (a) floorplan (b) assignment of modules to the two different Vdd domains.

Fig. 10(a) shows the floorplan of a microarchitecture used in [16] with ten voltage regulators inserted. The microarchitecture

Benchmark	Type	Instruction (B)
vpr	Integer	11
gcc	Integer	35
gzip	Integer	63
bzip2	Integer	94
parser	Integer	301
art	Floating-point	54
quake	Floating-point	175
mesa	Floating-point	305

TABLE I

BENCHMARKS FROM THE SPEC 2000 SUITE, ALONG WITH THE REFERENCE INSTRUCTION COUNTS IN BILLIONS.

is based on the DLX architecture [17] and the dark regions represent voltage regulators. We first simulate the architecture using the eight benchmark programs contained in the SPEC 2000 suite that cover both integer and floating-point operations. The eight programs with their respective instruction counts in billions are listed in Table I. To speed up the simulations, we utilize SMARTS [18], a periodic sampling technique to obtain the current consumption trace of each module for each of the benchmark programs. Specifically, we start simulating a program at clock cycle 0 and continue the simulation for s consecutive clock cycles. The average current consumption of each module is calculated during this period of time. Next, we stride forward and start the simulation again at the l^{th} clock cycle with $l \gg s$. The average current consumption of each module is calculated again for the s clock cycles that follow. This process continues until the entire program is completed. We can see clearly that by using this strategy, we obtain a sampled sequence of the average current consumption trace of each module.

Note that the time averaged total current consumption of the chip may vary significantly while running different programs, and the objective of our algorithm is to obtain a partition of modules that is deemed good across the entire benchmark suite, i.e., we want to ensure that each benchmark program imposes a similar weight in affecting the partition of modules. To achieve this objective, we normalize the current consumption traces associated with each program so that the normalized average total current consumption of the chip while running that program becomes 1. In Table II, we show an example that contains only two modules and for which the simulation result is collected at only two sample points. The total current consumption of the design at the two sample points are 30mA and 50mA, respectively. Therefore, the average total current consumption of the design is $(30\text{mA}+50\text{mA})/2 = 40\text{mA}$. We use 40mA to normalize the current consumptions of the two modules at the two sample points, which generates the unit-less numbers shown on the right of the table for the normalized current consumption traces. Next, we concatenate the normalized current consumption traces for all of the programs such that a combined current trace is obtained for each module. The combined current traces are used to build the

	I_{M_1}	I_{M_2}	I_{total}	$I_{M_1}^N$	$I_{M_2}^N$
Sample 1	10mA	20mA	30mA	0.25	0.5
Sample 2	20mA	30mA	50mA	0.5	0.75

TABLE II

AN EXAMPLE OF NORMALIZING THE CURRENT CONSUMPTION TRACES.

graph as described in Section III.B, and the graph is partitioned using the algorithm presented in Section III.C.

The result of the partition is shown in Fig. 10(b), where the lightly darkened regions represent the modules operating between the $2V_{dd}$ and V_{dd} rails while the white regions represent the modules operating between the V_{dd} and GND rails.

B. Experimental setup for the validation of the module assignment

To validate that the module assignment obtained by our algorithm indeed reduces the power wasted in voltage regulators, we first use a regular grid structure similar to that shown in Fig. 11(a) to represent the V_{dd} rail, and we assume that current sources are attached to the nodes in the power grid, which model the currents consumed by modules. Next, we associate each node in the power grid with a region on the chip and assume that all the modules located in that region only source(sink) currents from(to) that particular node, e.g., the dark region surrounding the black colored node in Fig. 11(a) is associated with that node. For node i with the associated region A_i , if only part of a module M overlaps with A_i , then only the corresponding portion of the current consumed by M is attached to node i . For example, in Fig. 11(b), only 25% of module M overlaps with the area associated with the power grid node. As a result, only 25% of the current consumed by module M is attached to that particular node.

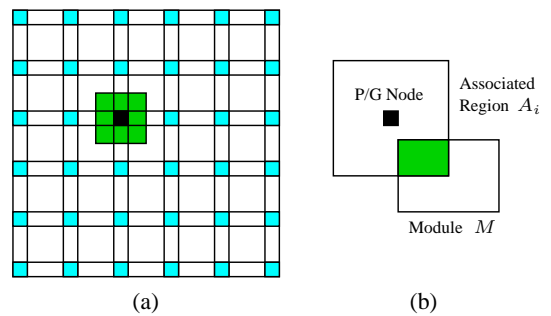


Fig. 11. Testing the actual wasted power (a) the structure of the V_{dd} grid and the region that source(sink) current from(to) a particular node (b) calculating the current source attached to a power grid node when the module only partially overlaps the region associated with the node.

After the value of the current source attached to each power grid node is calculated, a modified nodal analysis (MNA) equa-

tion in the form

$$\begin{pmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{pmatrix} \begin{pmatrix} V \\ I_{reg} \end{pmatrix} = \begin{pmatrix} I_s \\ V_{reg} \end{pmatrix} \quad (12)$$

can be established to calculate the current flowing through each of the voltage regulators, and therefore the wasted power. Here G_{ij} 's are submatrices of the coefficient matrix, V is the vector of nodal voltages, I_{reg} is the vector containing the currents flowing through voltage regulators, I_s is the vector of known current sources attached to the nodes in the V_{dd} grid, and V_{reg} is a vector whose components are all V_{dd} 's, the voltage level maintained by the regulators.

C. Result of comparison between different module assignment schemes for the DLX architecture

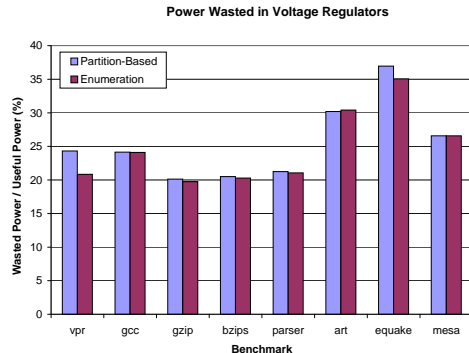
We have studied two different module assignment schemes for the DLX architecture shown in Section IV.A, one using the algorithm presented in this paper, and the other using an enumeration approach. For the enumeration method, we walk through all possible module assignments in the design space, and for each assignment, we use equation (2) to calculate the wasted current flowing through each of the regulators. The normalized current described in Section IV.A is used in the place of $I_i(t)$, and the sum of the time averaged wasted currents is calculated to characterize the power efficiency of the assignment. Since the DLX architecture contains only sixteen modules, enumeration can still be completed in a reasonable amount of time although it is much slower than the partition-based approach, and the result from enumeration provides a good criterion in judging how effective the partition-based approach is in assigning modules to different Vdd domains.

The experiments are carried out on a desktop with a 3.2GHz Pentium-4 CPU. It takes the partition-based approach 10msec to reach the final solution, while the enumeration requires about 230sec to exam all possible assignments in the design space.

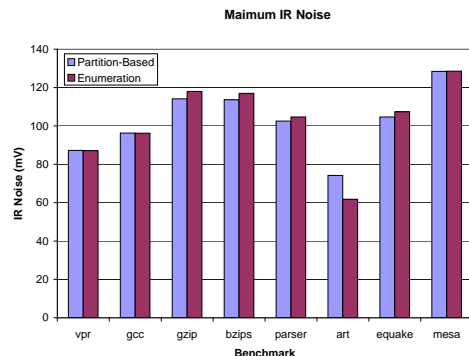
Fig. 12(a) shows the comparison of the power wasted in voltage regulators between the partition-based module assignment scheme and the enumeration approach using the validation method described in Section IV.B. In obtaining the figure, we divide the total power wasted in voltage regulators by the total power consumed by regular modules, where the latter is termed ‘‘useful power’’. We can see clearly that the power wasted in the design obtained using the partition-based approach is rather close to that in the best design generated through enumeration, although the former enjoys a significant advantage in terms of runtime.

It is important to notice that a design optimized for power also tends to achieve low IR noise in the V_{dd} grid. This is because to reduce the power waste, good current balance must be maintained locally as described in Section II, which is beneficial to reducing the IR noise since the current consumed by a module in one Vdd domain will immediately be recycled by some nearby modules in the other Vdd domain without flowing through a long resistive path in the power grid. In Fig. 12(b), we compare the maximum IR noise encountered in the V_{dd} grid when the module assignment

is performed using the two different schemes presented above, respectively. It can be seen that our design technique achieves IR noise levels that are comparable with those obtained through enumeration.



(a)



(b)

Fig. 12. Comparison between the module assignments using the partition-based approach and enumeration in terms of (a) the total power wasted in voltage regulators, and (b) the worst case IR noise in the V_{dd} grid.

D. Result of comparison between different module assignment schemes for 3D circuits

We have also studied the effectiveness of using the stacked-Vdd paradigm in 3D circuit designs, where each 3D circuit is assumed to have three active tiers. The GSRC benchmarks n100, n200, and n300 are chosen over the MCNC benchmarks in our experiment because the former contain more modules, which can

better reflect the complexity of the problems encountered in 3D designs. Module assignment is performed for each tier in the 3D circuit, and our partition-based assignment algorithm is compared with a simulated annealing technique.

For the simulated annealing method, a single module is switched to the opposite Vdd domain in each move since the entire design space can be explored by sequentially switching individual modules, and for each assignment under examination, the power wasted in voltage regulators is calculated using the same approach as in the enumeration method described in the previous sub-section. We choose simulated annealing over enumeration as the algorithm being compared with because the runtime of the enumeration method is exponential with respect to the number of modules in the floorplan, and for floorplans containing hundreds of modules, the enumeration method is no longer practical.

Because of the lack of available current traces for GSRC benchmarks, we have assumed that the mean current consumption of each module is a random number between 100mA and 1000mA, and the instantaneous current consumption of module M_i is assumed to be varying randomly around its mean I_i^{mean} .²

Table III shows the result of comparison between the module assignments obtained using our partition-based approach and those using the simulated annealing technique, in terms of the power wasted in voltage regulators, the maximum IR noise in the V_{dd} grid, and the time it takes to reach the assignments. The wasted power and the maximum IR noise are calculated using the validation method described in Section IV.B after the assignments have been obtained. We can see that the partition-based approach can generate module assignments with similar quality to those obtained through simulated annealing. However, the time it takes for the partition-based approach to find a good solution is orders of magnitude smaller.

V. CONCLUSIONS

In this paper, we presented a partition-based algorithm for efficiently assigning modules to different Vdd domains in a two level stacked-Vdd circuit. The objective is to minimize the power wasted in voltage regulators. The primary steps of the algorithm include building a graph that represents the correlations between modules and performing a max-cut partitioning of the graph using a F-M like algorithm. Experimental results on a DLX architecture and 3D circuit examples show that our module assignment algorithm can achieve designs with power and noise performance comparable to those obtained using enumeration and simulated annealing techniques, while at the same time, our method enjoys orders of magnitude speedup in runtime.

²The instantaneous current consumption of module M_i is given by $I_i(t) = I_i^{mean} \times (1 + \delta_g(t)) \times (1 + \delta_i(t))$, where $\delta_g(t)$ is a random number that remains the same for all of the modules, and $\delta_i(t)$ is a random number that varies from module to module. It is not difficult to see that $\delta_g(t)$ can be used to model the change of operations that affect the entire chip while $\delta_i(t)$ can be used to model the local variation in current consumption as a function of time t . In our experiment, $\delta_g(t)$ is randomly selected within the range $[-0.3, 0.3]$, and $\delta_i(t)$ is randomly selected within the range $[-0.2, 0.2]$.

APPENDIX

In this appendix, we prove that the module assignment problem formulated in Section III.A is NP-hard. The proof is based on the following known NP-complete set partitioning problem [19].

NP-complete set partitioning problem: Given a multiset S of integers, is there a way to partition S into two subsets S_1 and S_2 such that the sums of the numbers in each subset are equal.

Three problems will be shown to be NP-hard in tandem with each one being more comprehensive than the previous one, and the last one corresponding to the original module assignment problem shown in Section III.A.

Problem 1: Assume that there is a set of modules and a single regulator on the chip, and assume that the current flowing through the regulator is

$$I_R = \left| \sum_{j \in 2V_{dd}} I_j - \sum_{j \in V_{dd}} I_j \right| \quad (13)$$

find an assignment of modules such that the current flowing through the regulator is minimized.

It is obvious that Problem 1 is the same as Problem 1A shown below.

Problem 1A: Given a set \mathcal{I} of numbers $\{I_1, I_2, \dots\}$ (not necessarily integers), find a partition of \mathcal{I} into two subsets \mathcal{I}_1 and \mathcal{I}_2 such that $\left| \sum_{j \in \mathcal{I}_1} I_j - \sum_{j \in \mathcal{I}_2} I_j \right|$ is minimized.

It is easy to see that the NP-complete set partitioning problem can be reduced to Problem 1A because if we set \mathcal{I} in Problem 1A to S in the set partitioning problem, then there is a way to partition the set S if and only if the minimum value of $\left| \sum_{j \in \mathcal{I}_1} I_j - \sum_{j \in \mathcal{I}_2} I_j \right|$ in Problem 1A is zero. Therefore, Problem 1A is NP-hard. Since problem 1 is completely the same as Problem 1A, Problem 1 is also NP-hard.

Problem 2: Assume that there are a set of modules and K regulators on the chip, and assume that the modules in a region only interact with the regulator in the same region, find a assignment of modules such that the sum of the currents flowing through the regulators is minimized.

Problem 1 can be reduced to Problem 2 through the following procedure. We make a chip that contains K regulators and K groups of modules, where each group of modules corresponds to one copy of the non-regulator modules in Problem 1. We can further assume that each group of modules only interact with one regulator. The partition of each group of modules will optimally solve Problem 1 because otherwise, we can replace the group that does not optimally solve Problem 1 by the optimal solution

Layer	$\frac{\text{WastedPower}}{\text{UsefulPower}} (\%)$		Maximum IR Noise (mV)		Runtime (sec)	
	Partition-Based	Annealing	Partition-Based	Annealing	Partition-Based	Annealing
n100Layer0	3.3	3.1	52.8	62.0	0.03	80
n100Layer1	3.1	3.8	28.9	42.5	0.02	80
n100Layer2	3.7	5.7	45.4	54.6	0.02	80
n200Layer0	8.7	6.4	55.2	88.4	0.31	157
n200Layer1	5.6	6.4	62.1	64.4	0.16	160
n200Layer2	5.6	7.1	77.4	52.7	0.18	165
n300Layer0	4.7	4.5	61.1	56.0	1.83	235
n300Layer1	6.3	6.3	33.4	36.8	0.69	236
n300Layer2	5.4	4.6	46.5	39.5	0.77	236

TABLE III

COMPARISON BETWEEN THE MODULE ASSIGNMENT OBTAINED USING THE GRAPH PARTITION-BASED APPROACH AND THAT USING THE SIMULATED ANNEALING TECHNIQUE.

of Problem 1, which will reduce the total current flowing through the regulators. Since Problem 1 has been proven to be NP-hard, Problem 2 is also NP-hard.

Problem 3: Assume that there are a set of modules and K regulators on the chip, and assume that the modules in a region only interact with the regulator in the same region, find a assignment of modules such that the sum of the time-averaged currents flowing through the regulators is minimized.

Problem 3 is completely equivalent to the module assignment problem formulated in Section III.A, and it differs from Problem 2 only in that the summation is over the time-averaged current flowing through each of the regulators. It can be seen that Problem 2 can be reduced to Problem 3 because Problem 2 corresponds to a special case of Problem 3 where the current flowing through each module does not vary with time. Since Problem 2 has been proven to be NP-hard, Problem 3 is also NP-hard.

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 system-on-chip integration," *Proceedings of the IEEE*, vol. 89, pp. 602–633, May 2001.
- [2] W. R. Davis, J. Wilson, S. Mick, J. Xu, H. Hua, C. Mineo, A. M. Sule, M. Steer, and P. D. Franzon, "Demystifying 3D ICs: The pros and cons of going vertical," *IEEE Design and Test of Computers*, vol. 22, pp. 498–510, Nov.-Dec. 2005.
- [3] J. L. Tsai, C. C. P. Chen, G. Chen, B. Goplen, H. Qian, Y. Zhan, S. M. Kang, M. D. F. Wong, and S. S. Sapatnekar, "Temperature-aware placement for SOCs," *Proceedings of the IEEE*, vol. 94, pp. 1502–1518, Aug. 2006.
- [4] Y. Zhan and S. S. Sapatnekar, "High-efficiency Green function-based thermal simulation algorithms," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, pp. 1661–1675, Sept. 2007.
- [5] P. Li, L. T. Pileggi, M. Asheghi, and R. Chandra, "IC thermal simulation and modeling via efficient multigrid-based approaches," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, pp. 1763–1776, Sept. 2006.
- [6] B. Goplen and S. S. Sapatnekar, "Placement for 3D ICs with thermal and interlayer via considerations," in *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 626–631, Jun. 2007.
- [7] B. Goplen and S. S. Sapatnekar, "Efficient thermal placement of standard cells in 3D ICs using a force directed approach," in *Digest of Technical Papers, IEEE/ACM International Conference on Computer-Aided Design*, pp. 86–89, Nov. 2003.
- [8] B. Goplen and S. S. Sapatnekar, "Placement of thermal vias in 3-D ICs using various thermal objectives," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, pp. 692–709, Apr. 2006.
- [9] J. Cong, G. Luo, J. Wei, and Y. Zhang, "Thermal-aware 3D IC placement via transformation," in *Proceedings of the IEEE Asia and South Pacific Design Automation Conference*, pp. 780–785, Jan. 2007.
- [10] J. Cong, J. Wei, and Y. Zhang, "A thermal-driven floorplanning algorithm for 3D ICs," in *Digest of Technical Papers, IEEE/ACM International Conference on Computer-Aided Design*, pp. 306–313, Nov. 2004.
- [11] Semiconductor Industry Association, "International technology roadmap for semiconductors, 2006." <http://public.itrs.net/Links/2006Update/2006UpdateFinal.htm>.
- [12] S. Rajapandian, K. Shepard, P. Hazucha, and T. Karnik, "High-tension power delivery: Operating 0.18 μm CMOS digital logic at 5.4V," in *Digest of Technical Papers, IEEE International Solid-State Circuits Conference*, pp. 298–299, Feb. 2005.
- [13] J. Gu and C. H. Kim, "Multi-story power delivery for supply noise reduction and low voltage operation," in *Proceedings of the International Symposium on Low Power Electronics and Design*, pp. 192–197, Aug. 2005.
- [14] E. Chiprout, "Fast flip-chip power grid analysis via locality and grid-shells," in *Digest of Technical Papers, IEEE/ACM International Conference on Computer-Aided Design*, pp. 485–488, Nov. 2004.
- [15] S. M. Sait and H. Youssef, *VLSI Physical Design Automation: Theory and Practice*. Piscataway, NJ: IEEE Press, 1995.
- [16] V. Nookala, D. Lilja, and S. S. Sapatnekar, "Temperature-aware floorplanning of microarchitecture blocks with IPC-power dependence modeling and transient analysis," in *Proceedings of the International Symposium on Low Power Electronics and Design*, pp. 298–303, Oct. 2006.
- [17] D. A. Patterson and J. L. Hennessy, *Computer Architecture: A Quantitative Approach*. San Francisco, CA: Morgan Kaufmann Publishers, 2nd ed., 1996.
- [18] R. E. Wunderlich, T. F. Wenisch, B. Falsafi, and J. C. Hoe, "SMARTS: Accelerating microarchitecture simulation via rigorous statistical sampling," in *Proceedings of the Annual International Symposium on Computer Architecture*, pp. 84–97, Jun. 2003.
- [19] R. M. Karp, "Reducibility among combinatorial problems." in R. E. Miller and J. W. Thatcher (editors): *Complexity of Computer Computations*. New York: Plenum, pp. 85–103, 1972.