Minimum Implant Area-Aware Placement and Threshold Voltage Refinement

Seong-I Lei
Department of Computer Science
National Tsing Hua University
Hsinchu, Taiwan 300 R.O.C.
d9762804@oz.nthu.edu.tw

Wai-Kei Mak
Department of Computer Science
National Tsing Hua University
Hsinchu, Taiwan 300 R.O.C.
wkmak@cs.nthu.edu.tw

Chris Chu
Department of Electrical and Computer Engineering
Iowa State University
Ames, Iowa 50011
cnchu@iastate.edu

Abstract— Threshold voltage assignment is a very effective technique to reduce leakage power consumption in modern integrated circuit (IC) design. As feature size continues to decrease, the layout constraints (called MinIA constraints) on the implant area, which determines the threshold voltage of a device, are becoming increasingly difficult to satisfy. It is necessary to take these constraints into consideration during the layout stage. In this paper, we propose to resolve the MinIA constraint violations by a simultaneous detailed placement and threshold voltage refinement approach. We present an optimal and efficient mixed integer-linear programming (MILP)-based algorithm which guarantees to fix all MinIA constraint violations. Experimental results demonstrate that our algorithm only perturbs the original placement and threshold voltage assignment solutions minimally to eliminate all violations and is fast in practice.

I. INTRODUCTION

Multiple threshold voltages (multi-V_t) are commonly employed in advanced power-aware high-performance chip design. For example, a chip design can make use of standard cells operating at standard V_t (SVT), low V_t (LVT) and ultra-low V_t (ULVT). Cells operating at lower threshold voltage are faster but more leaky. So, to attain high performance while controlling the leakage power, designers can use cells with lower V_t on critical timing paths and cells with higher V_t on non-critical paths.

There are physical constraints associated with the threshold voltage assignment. During fabrication, local V_t implant areas can be formed at different locations of a chip. The V_t of a placed cell is determined by the ion implantation of the implant area that it belongs to. There are some manufacturing restrictions for the V_t implant areas called the minimum implant area (MinIA) constraints [1–3]. First, each V_t implant area must have a certain minimum width. Second, two V_t implant areas of the same type must be separated by a certain minimum spacing. The MinIA constraints are illustrated in Figure 1, which shows a standard cell row with four standard cells. Cells a and b are of one threshold voltage while cells c and d are of another voltage. Cell d is not wide enough and hence it will cause a minimum implant width violation. Cells c and d are of the same type and are too close to each other. Hence they will cause a minimum implant spacing violation. Note that although the spacing between cells b and c are less

than the required minimum implant spacing, they are of different threshold voltage and so will not cause any minimum spacing violation.
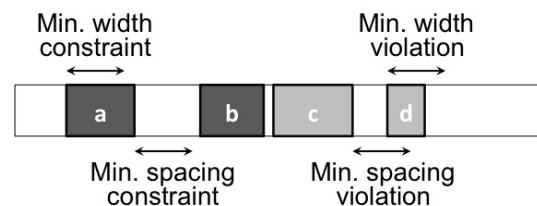

Fig. 1. Minimum implant width and spacing constraints.

There are many works on multiple threshold voltage assignment, e.g., [4–7]. However, those works did not consider the MinIA constraints. Consequently, different threshold voltages may be assigned to neighboring cells that results in implant areas violating the MinIA constraints.

Filler cell insertion is a common way to fix MinIA violations as illustrated in Figure 2. For a narrow V_t-island, if there is sufficient whitespace adjacent to it, then a filler cell of the same implant type can be inserted into the whitespace to enlarge the V_t-island to satisfy the minimum width constraint (as shown in Figure 2(a)). Two V_t-islands of the same type separated by a small whitespace can be merged into one island by inserting a filler cell of the same implant type into the whitespace to eliminate the minimum spacing constraint violation (as shown in Figure 2(b)). Filler cell insertion is supported by commercial tools to fix the minimum implant area violation. For instance, Synopsys IC Compiler [8] offers a threshold-voltage-aware filler cell insertion flow which allows the user to define the V_t filler cell to be inserted into each whitespace according to the threshold voltages of the cells on the two sides of the whitespace. For example, the user can specify rules to always insert LVT filler cell in the whitespace between LVT and SVT cells, and always insert ULVT filler cell in the whitespace between ULVT and LVT cells or between ULVT and SVT cells.

In the past, MinIA constraints are not an important design consideration as MinIA violations are relatively easy to fix by filler cell insertion. However, as feature size keep diminishing, the critical features are printed by multiple patterning or other expensive processes while the less-critical implant areas are not. Thus the implant areas scale at a slower rate and
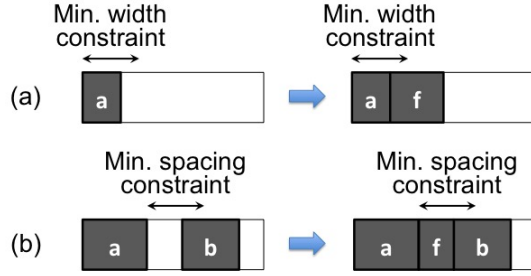
that there is an allowable displacement range for each cell depending on its timing criticality. A non-timing critical cell may have a large allowable displacement range while a highly timing critical cell may have a zero allowable displacement range. Moreover, the V_t of each cell is allowed to be reduced but not raised to avoid timing violation.

III. ROW-BASED OPTIMAL ALGORITHM

We propose a row-based approach that take all the cells in a row and perform optimization to eliminate all minimum implant area rule violations in the row at once. This is unlike existing practice which tries to fix the minimum implant area constraint violations one by one using local greedy heuristics. We present an optimal algorithm to solve the MinIA-aware detailed placement and V_t refinement problem using mixed integer linear programming (MILP).

III.A. Useful Observations

Before presenting the MILP model, we first make some useful observations which we are going to take advantage of. First, the heuristic in [1] always inserts just a single type of filler cell into the whitespace between two regular cells. But we note that a better approach is to divide the whitespace into a left subspace and a right subspace (a subspace can be empty) such that each is inserted with its own desired filler cell type. For example, in Figure 4, inserting just a single filler cell type into the whitespace between cell a and b can fix either the MinIA width violations on the left or on the right of the whitespace as in Figure 4(b)(c). However, dividing the whitespace into two subspaces and insert two filler cell types as in Figure 4(d) will fix both MinIA width violations on the left and on the right at the same time. And it is easy to see that the left (right) subspace should always be inserted with filler cell type matching the implant type of the cell on its left (right). So, we propose a MILP model that will automatically determine the optimal division for each whitespace assuming that the left (right) subspace is always inserted with filler cell type matching the implant type of the cell on its left (right).
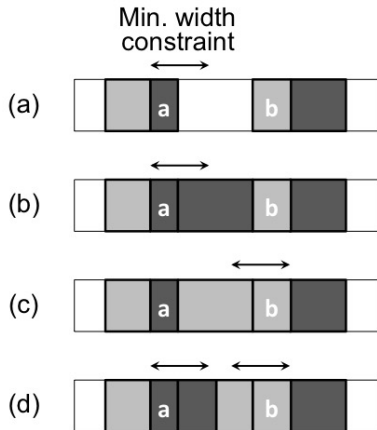
Fig. 4. A better approach of filler cell insertion.

Second, the minimum spacing S is no larger than the minimum width W in practice. So, if a cell row is made up of abut-

ting implant areas all satisfying the minimum width constraint with no empty gap, then the minimum spacing constraint for the same implant type will be automatically satisfied. It is because two implant areas of the same type will have at least one implant area of a different type in between them, hence their separation will be at least $W (\geq S)$. As explained in the paragraph above, our proposed MILP model assumes that all whitespaces in a row will be inserted with filler cells leaving no empty gap in the row, thus it is sufficient to consider minimum implant width constraint only.

III.B. MILP Model

We assume that the cells in a row are indexed from 1 to n from the left to the right. We assume that there are three levels of threshold voltages which are SVT, LVT, and ULVT, though our model can be easily extended to any number of threshold voltage levels. Below are the inputs to the MinIA-aware detailed placement and V_t refinement problem.

- n: number of cells in the row.

- L: row length.

- W: minimum implant width.

- W_k: width of cell k.

- X_k: original location of cell k's left boundary.

- $\delta_k^{min}, \delta_k^{max}$: [$\delta_k^{min}, \delta_k^{max}$] is the range of allowed displacement for cell k ($\delta_k^{min} \leq 0$ and $\delta_k^{max} \geq 0$).

- $\Delta_k^{V_s}, \Delta_k^{V_l}, \Delta_k^{V_{ul}}$: power penalties if cell k's threshold voltage is re-assigned to SVT, LVT, ULVT, respectively.

The outputs of the MinIA-aware detailed placement and V_t refinement problem are as follows.

- $a_{k,s}, a_{k,l}, a_{k,ul}$: 1 if cell k's threshold voltage is re-assigned to SVT, LVT, ULVT, respectively; 0 otherwise.

- δ_k: final displacement of cell k from its original location.

- m_k: an intermediate location between the final locations of cells k and $k+1$ that divides the whitespace between cells k and $k+1$ into two subspaces such that the left (right) subspace is inserted with filler cell type matching the implant type of cell k ($k+1$).

Our MILP model is given below.

$$\text{Min. } \alpha \sum_k \left(\Delta_k^{V_s} \cdot a_{k,s} + \Delta_k^{V_l} \cdot a_{k,l} + \Delta_k^{V_{ul}} \cdot a_{k,ul} \right) + \beta \sum_k \bar{\delta}_k$$

$$\text{s.t. } a_{k,s} + a_{k,l} + a_{k,ul} = 1 \ \forall k \tag{1}$$

$$a_{k,\tau} = 0 \ \forall k, \tau \text{ s.t.}$$
$$\text{cell } k \text{ cannot be assigned threshold voltage } \tau \tag{2}$$

$$a_{k,s}, a_{k,l}, a_{k,ul} = 0 \text{ or } 1 \ \forall k \tag{3}$$

$$\delta_k^{min} \leq \delta_k \leq \delta_k^{max} \ \forall k \tag{4}$$

$$\bar{\delta}_k \geq \delta_k \ \forall k \tag{5}$$

$$\bar{\delta}_k \geq -\delta_k \ \forall k \tag{6}$$

$$X_k + \delta_k + W_k \leq m_k \leq X_{k+1} + \delta_{k+1}$$

$$\forall 1 \le k \le n-1 \tag{7}$$
$$0 \le X_1 + \delta_1 \tag{8}$$
$$X_n + \delta_n + W_n \le L \tag{9}$$
$$d_{k,k+1} \le 1 - a_{k+1,s} + 1 - a_{k,s} \ \ \forall 1 \le k \le n-1 \tag{10}$$
$$d_{k,k+1} \ge a_{k+1,s} - a_{k,s} \ \ \forall 1 \le k \le n-1 \tag{11}$$
$$d_{k,k+1} \le 1 - a_{k+1,l} + 1 - a_{k,l} \ \ \forall 1 \le k \le n-1 \tag{12}$$
$$d_{k,k+1} \ge a_{k+1,l} - a_{k,l} \ \ \forall 1 \le k \le n-1 \tag{13}$$
$$d_{k,k+1} \le 1 - a_{k+1,ul} + 1 - a_{k,ul} \ \ \forall 1 \le k \le n-1 \tag{14}$$
$$d_{k,k+1} \ge a_{k+1,ul} - a_{k,ul} \ \ \forall 1 \le k \le n-1 \tag{15}$$
$$d_{k,k+1} = 0 \text{ or } 1 \ \ \forall 1 \le k \le n-1 \tag{16}$$
$$m_{k+j} - m_{k-1} \ge (d_{k-1,k} + d_{k+j,k+j+1} - 1)W$$
$$\forall 1 \le k \le n \text{ and}$$
$$\forall j \ge 0 \text{ s.t. } k+j \le n \text{ and } \sum_{r=0}^{j} W_{k+r} < W \tag{17}$$
$$d_{0,1} = 1 \tag{18}$$
$$d_{n,n+1} = 1 \tag{19}$$
$$m_0 = 0 \tag{20}$$
$$m_n = L \tag{21}$$

The objective of the MILP is to minimize a weighted sum of power overhead and total cell displacement. The power overhead is given by $\sum_k (\Delta_k^{V_s} \cdot a_{k,s} + \Delta_k^{V_l} \cdot a_{k,l} + \Delta_k^{V_{ul}} \cdot a_{k,ul})$ while the total displacement of all cells is given by $\sum_k \bar{\delta}_k$. α and β are user-defined weighting constants. Constraints (1)-(3) ensure that each cell is re-assigned to a legal threshold voltage. In particular, we use constraint (2) to forbid any cell to be re-assigned to a threshold voltage higher than its initial threshold voltage to avoid timing violation. Constraint (4) bounds the displacement for each cell based on the given allowed displacement range. As a negative displacement represents a displacement to the left and a positive displacement represents a displacement to the right, we use constraints (5)-(6) to get a tight lower bound for the magnitude of the displacement of cell k denoted by $\bar{\delta}_k$. Moreover, since $\bar{\delta}_k$ appears in the minimization objective, it will force $\bar{\delta}_k$ to be exactly equal to the magnitude of the displacement of cell k. Constraint (7) ensures that the cells in the row will not overlap and the division point of the whitespace between any two adjacent cells must be legal, i.e., between the two cells' final locations. Constraints (8)-(9) ensure that the leftmost and the rightmost cells must lie within the given placement region.

We introduce binary variable $d_{k,k+1}$ which indicates if cells k and $k+1$ are re-assigned distinct threshold voltages or not. Therefore, we want $d_{k,k+1}$ to be equal to 1 if and only if cells k and $k+1$ are re-assigned distinct threshold voltages. It is accomplished by constraints (10)-(16). If cells k and $k+1$ are re-assigned distinct threshold voltages, then the conjunction of constraints (11), (13) and (15) is equivalent to $d_{k,k+1} \ge 1$ while the conjunction of constraints (10), (12), and (14) is equivalent to $d_{k,k+1} \le 1$, hence $d_{k,k+1}$ will be 1. On the other hand, if cells k and $k+1$ are re-assigned the same threshold voltage, then the conjunction of constraints (11), (13) and (15) is equivalent to $d_{k,k+1} \ge 0$ while the conjunction of constraints (10), (12), and (14) is equivalent to $d_{k,k+1} \le 0$, hence $d_{k,k+1}$ will be 0. If $\sum_{r=0}^{j} W_{k+r} < W$, then it is possible that cells k to $k+j$ would form a V_t-island that violates the minimum width constraint. To ensure

that it will never happen, we impose constraint (17) whenever $\sum_{r=0}^{j} W_{k+r} < W$. Note that cells k to $k+j$ will form a V_t-island only if cell k's V_t is different from that of cell $k-1$ (i.e., $d_{k-1,k} = 1$) and cell $k+j$'s V_t is different from that of cell $k+j+1$ (i.e., $d_{k+j,k+j+1} = 1$). In that case, constraint (17) will become $m_{k+j} - m_{k-1} \ge W$ enforcing that such island is large enough. And constraint (17) is trivially satisfied unless both $d_{k-1,k}$ and $d_{k+j,k+j+1}$ are 1. Finally, constraints (18)-(21) take care of the boundary conditions.

We note that our MILP is guaranteed to be feasible since $a_{k,ul} = 1$ for all k, $\delta_k = 0$ for all k, and $m_k = X_{k+1}$ for $k = 1, \ldots, n-1$ is always a feasible solution to it. That is, one way (not the best way) to resolve all MinIA constraint violations is to re-assign all cells' threshold voltage to ULVT and fill all whitespace with ULVT filler cells without moving any cell. Our approach always tries to find the best feasible solution to fix all MinIA constraint violations with the minimum power overhead and total cell displacement.

III.C. A Simpler and Faster MILP Model

In this subsection, we discuss a possible simplification to our proposed MILP. Instead of using binary variable $d_{k,k+1}$ which indicates if cells k and $k+1$ are re-assigned distinct threshold voltages or not, we may introduce binary variable $d'_{k,k+1}$ such that $d'_{k,k+1}$ must be 1 if cells k and $k+1$ are re-assigned distinct threshold voltages. And we may replace constraints (10)-(19) by constraints (22)-(28) below. If cells k and $k+1$ are re-assigned distinct threshold voltages, then the conjunction of constraints (22), (23) and (24) is equivalent to $d'_{k,k+1} \ge 1$, hence $d'_{k,k+1}$ will be 1. Otherwise, the value of $d'_{k,k+1}$ can be freely set by the MILP solver.

$$d'_{k,k+1} \ge a_{k+1,s} - a_{k,s} \ \ \forall 1 \le k \le n-1 \tag{22}$$
$$d'_{k,k+1} \ge a_{k+1,l} - a_{k,l} \ \ \forall 1 \le k \le n-1 \tag{23}$$
$$d'_{k,k+1} \ge a_{k+1,ul} - a_{k,ul} \ \ \forall 1 \le k \le n-1 \tag{24}$$
$$d'_{k,k+1} = 0 \text{ or } 1 \ \ \forall 1 \le k \le n-1 \tag{25}$$
$$m_{k+j} - m_{k-1} \ge (d'_{k-1,k} + d'_{k+j,k+j+1} - 1)W$$
$$\forall 1 \le k \le n \text{ and}$$
$$\forall j \ge 0 \text{ s.t. } k+j \le n \text{ and } \sum_{r=0}^{j} W_{k+r} < W \tag{26}$$
$$d'_{0,1} = 1 \tag{27}$$
$$d'_{n,n+1} = 1 \tag{28}$$

As explained previously, suppose $\sum_{r=0}^{j} W_{k+r} < W$, then we need to make sure that $m_{k+j} - m_{k-1} \ge W$ only if cell k's V_t is different from that of cell $k-1$ and cell $k+j$'s V_t is different from that of cell $k+j+1$. But we know that when cell k's V_t is different from that of cell $k-1$ and cell $k+j$'s V_t is different from that of cell $k+j+1$, both $d'_{k,k+1}$ and $d'_{k+j,k+j+1}$ will be equal to 1, so constraint (26) will become $m_{k+j} - m_{k-1} \ge W$ as desired. Otherwise, the MILP solver is free to set the value of at least one of $d'_{k,k+1}$ or $d'_{k+j,k+j+1}$ to satisfy constraint (26) trivially.

III.D. Dealing with Fixed Macros

In practice, there can be many fixed macros in the given placement that cannot be moved. Our approach can be eas-

2C-4

TABLE I
BENCHMARKS CHARACTERISTICS

	#Cells	Utilization (%)	% of narrow cells
superblue1	847K	69	67
superblue3	920K	73	45
superblue4	600K	70	54
superblue5	772K	77	47
superblue7	1.36M	76	34
superblue10	1.2M	70	64
superblue16	699K	69	59
superblue18	1.27M	67	40

TABLE III
RUNTIME REDUCTION WITH THE PROPOSED SIMPLIFICATION.

	Non-simplified MILP	Simplified MILP
	CPU time (s)	CPU time (s)
superblue1	162.21	146.29
superblue3	165.50	99.53
superblue4	103.31	69.66
superblue5	119.47	65.84
superblue7	243.52	124.72
superblue10	215.16	128.12
superblue16	141.64	98.02
superblue18	86.95	45.37
ratio	1.59	1

ily adapted to handle that. We just need to scan the design once and divide each row of cells into sub-rows separated by the fixed macros. Then we can formulate an MILP for each sub-row with the left end point and the right end point of the sub-row as the boundaries when refining the placement of the cells in the sub-row. Hence, we form and solve an independent MILP for each sub-row.

IV. EXPERIMENTAL RESULTS

We adopt the circuits from the ICCAD 2012 placement contest benchmark suite where each circuit has a set of fixed macros and a set of movable cells for our experiments. We ran a routability-driven placer [12] to obtain the initial placement for each circuit. Some key characteristics of the benchmarks are listed in Table 1. Percentage of narrow cells denotes the proportion of cells with width less than the minimum implant width constraint in a circuit. Besides, the ratio of cells assigned to SVT/LVT/ULVT initially is roughly 7:2:1. We set the allowable displacement for a cell according to its timing criticality. Cells assigned to SVT, LVT, and ULVT initially are assumed to be non-timing-critical, more timing-critical, and most timing-critical, respectively. The allowable displacement for non-timing-critical cells in our experiment is no more than ten placement sites as in [1]. Unless otherwise stated, the simpler MILP model in Section III.C was used to obtain the results reported in this section. All experiments were conducted on a 3.3GHz Linux machine with 128GB memory. Gurobi [13] was used to solve the MILPs.

As a baseline for comparison, we first found the minimum power overhead to fix all MinIA constraint violations by optimal threshold voltage re-assignment and filler cell insertion without any placement perturbation. The baseline results were obtained by imposing a zero allowable displacement range for all cells in our MILP. Note that it is equivalent to setting $\alpha = 1$ and $\beta = \infty$. The results are shown in the left part of Table 2. Δpower denotes the increase in leakage power due to threshold voltage re-assignment compared to the original configuration. It should be noted that Δpower is non-zero for all circuits, it means that filler cell insertion alone (even with the intelligent division of each whitespace for filler cell insertion presented in Section III.A) is unable to fix all MinIA constraint violations and hence some cells must have their threshold voltages reduced in order to fix all violations. In general, Δpower is larger for circuits with higher proportion of narrow cells.

Then we tried different settings for the weighting constants α and β in the objective function of our MILP. First, we tried $\alpha = 1$ and $\beta = 0$, which means we do not try to minimize the total cell displacement as long as each cell is moved within its allowable range. The results are shown in the middle part of Table 2. Displacement per cell, which is the average number of placement sites that a cell has moved from its original location, is reported. For reference, the minimum cell width in all designs is two placement sites. It can be seen that cell movement, or equivalently, whitespace re-distribution can facilitate filler cell insertion to fix MinIA constraint violation, so the power overhead can be reduced by 23.1% on average compared to the baseline. The average displacement per cell is less than 1 site for all benchmarks. Second, we adjusted the values of α and β to minimize the power overhead and total cell displacement at the same time. The results are shown in the right part of Table 2. It can be seen that cell displacement can be effectively reduced by $11\times$ while maintaining a similar level of power overhead reduction. Finally, we compare in Table 3 the runtime of solving the non-simplified MILP model against the simplified MILP model for the last experiment above. The runtime is about 1.6 times faster with the simplification proposed in Section III.C.

V. CONCLUSIONS

MinIA constraints have become more difficult to satisfy as average cell size continues to decrease in advanced processes. We considered a MinIA-aware detailed placement and threshold voltage refinement problem for low power chip design utilizing multiple threshold voltages. An optimal row-based algorithm based on mixed-integer linear programming was proposed. Cell movement, intelligent whitespace division for filler cell insertion, and threshold voltage re-assignment are considered simultaneously in our algorithm. Experimental results showed that with limited total cell displacement, the power overhead for fixing all MinIA rule violations can be reduced significantly.

TABLE II
COMPARISON OF POWER OVERHEAD AND TOTAL CELL DISPLACEMENT FOR FIXING ALL MINIA CONSTRAINT VIOLATIONS.

	Placement perturbation disallowed ($\alpha = 1, \beta = \infty$)			Placement perturbation not optimized ($\alpha = 1, \beta = 0$)			Balanced placement perturbation ($\alpha = 1, \beta = 0.2$)		
	Δpower (%)	Displacement per cell	CPU time (s)	Δpower (%)	Displacement per cell	CPU time (s)	Δpower (%)	Displacement per cell	CPU time (s)
superblue1	22.86	0	81.68	17.99	0.70	105.77	18.03	0.08	146.29
superblue3	14.58	0	66.28	10.99	0.79	78.55	11.02	0.06	99.53
superblue4	19.32	0	50.21	15.32	0.62	55.29	15.35	0.07	69.66
superblue5	15.57	0	60.73	12.20	0.72	59.57	12.23	0.06	65.84
superblue7	13.10	0	107.19	10.34	0.69	111.10	10.37	0.05	124.72
superblue10	16.39	0	101.48	11.16	0.86	103.64	11.18	0.08	128.12
superblue16	20.81	0	71.11	16.46	0.79	89.39	16.51	0.09	98.02
superblue18	13.26	0	43.72	10.04	0.33	41.64	10.06	0.02	45.37
ratio	1	-	1	0.769	1	1.11	0.771	0.09	1.34

REFERENCES

[1] A.B. Kakhng and H. Lee. Minimum implant area-aware gate sizing and placement. In *Proc. ACM Great Lakes Symposium on VLSI*, pages 57–62, 2014.

[2] Andrew B Kahng. Lithography-induced limits to scaling of design quality. In *Proceedings of SPIE Advanced Lithography*, pages 905302–905302. International Society for Optics and Photonics, 2014.

[3] Andrew B Kahng. New game, new goal posts: A recent history of timing closure. In *Proc. ACM/IEEE Design Automation Conference*, 2015.

[4] Vijay Sundararajan and Keshab K Parhi. Low power synthesis of dual threshold voltage CMOS VLSI circuits. In *Proc. International Symposium on Low Power Electronics and Design*, pages 139–144. ACM, 1999.

[5] Pankaj Pant, Rabindra K Roy, and A Chattejee. Dual-threshold voltage assignment with transistor sizing for low power CMOS circuits. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 9(2):390–394, 2001.

[6] Mahesh Ketkar and Sachin S Sapatnekar. Standby power optimization via transistor sizing and dual threshold voltage assignment. In *Proc. IEEE/ACM International Conference on Computer Aided Design*, pages 375–378. IEEE, 2002.

[7] Yifang Liu and Jiang Hu. A new algorithm for simultaneous gate sizing and threshold voltage assignment. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 29(2):223–234, 2010.

[8] Synopsys IC Compiler user guide. http://www.synopsys.com/Tools/Implementation/PhysicalImplementation/Pages/ICCompiler.aspx.

[9] P. Gupta, A.B. Kahng, and C.H. Park. Detailed placement for improved depth of focus and CD constrol. In *Proc. Asia and South Pacific Design Automation Conference*, pages 343–348, 2005.

[10] H. Tian, Y. Du, H. Zhang, Z. Xiao, and D.F. Wong. Triple patterning aware detailed placement with constrained pattern assignment. In *Proc. IEEE/ACM International Conference on Computer Aided Design*, pages 116–123, 2014.

[11] T. Lin and C. Chu. TPL-aware displacement-driven detailed placement refinement with coloring constraints. In *Proc. ACM International Symposium on Phyiscal Design*, pages 75–80, 2015.

[12] Tao Lin and Chris Chu. POLAR 2.0: An effective routability-driven placer. In *Proc. ACM/IEEE Design Automation Conference*. IEEE, 2014.

[13] Gurobi. http://www.gurobi.com.