Patterning Aware Design Optimization of Selective Etching in N5 and Beyond

Yibo Lin[1, 2], Peter Debacker[2], Darko Trivkovic[2], Ryoung-Han Kim[2], Praveen Raghavan[2], and David Z. Pan[1]

[1]ECE Department, University of Texas at Austin, Austin, TX, USA
[2]IMEC, Leuven, Belgium
yibolin@utexas.edu, {pdback, Darko.Trivkovic, Ryan.Ryoung.han.Kim, Praveen.Raghavan}@imec.be, dpan@ece.utexas.edu

***Abstract*—The aggressive scaling towards N5 causes the difficulty in controlling process variations for line and block (cut) printing, while the margin for edge placement errors (EPE) is decreasing. Selective etching, implemented by self-aligned multiple patterning with alternating materials, introduces better EPE margins with self-aligned blocks (SAB). In this paper, we propose a systematic study on design space for SAB with various options of design rules. We also develop a post SAB optimization framework to evaluate the impacts of different rules. Based on the experiments on 6-track industrial benchmarks, we explore possible options for closure of current physical design flow.**

I. Introductions

Industry has been looking for how to define 5nm technology node (N5) and beyond with manufacturable process and acceptable cost. Multiple patterning options are considered for lower metal pitch in N5 which probably scales down to a range between 36nm and 20nm [1]–[3], e.g., 32nm for M2. This work considers extremely regular 1-D gridded layout style, where metal lines are first printed and unwanted parts are then trimmed off by cut masks. In the damascene process, metal lines are filled into trenches where blocks are used to stop filling unwanted parts. To avoid ambiguity, we use the terminology *block* instead of *cut*. Viable lithography options for 1-D layout in N5 include self-aligned multiple patterning (SAMP) for lines and multiple patterning lithography (MPL) or extreme ultraviolet (EUV) for blocks.

To fabricate the lines in Fig. 1(a), conventional blocks in Fig. 1(b) can only achieve a margin of $\frac{1}{4}P$ (P is metal pitch) to tolerate edge placement errors (EPE) from process variations by keeping the block end in the middle of two lines. To clarify, the *EPE margin* here refers to the maximum tolerable edge placement error between the design target and the manufactured result. When the pitch scales down, this margin becomes too small to guarantee enough yield in advanced nodes.

Self-aligned blocks (SAB) which incorporate selective etching with alternating-material SAMP significantly relieve the EPE margin issue [4]. The SAB implementation of lines in Fig. 1(d) can improve the EPE margin to $\frac{3}{4}P$ by alternating materials A and B for lines, where blocks for one material do not trim off the other. Furthermore, it allows cross-line merging of blocks which render more flexibility in designs, shown as long blocks in Fig. 1(d).

Previous work includes the study of yield model for SAB [4], various optimization techniques on conventional block distribution [4]–[8], and SAB decomposition [9]. In this paper, we explore the viability and design space of SAB in N5 and beyond by tweaking design rules for SAB and post SAB optimization for solution space exploration. Our major contributions are summarized as follows.

- We propose a set of design rules for SAB to enable more design flexibility without degrading manufacturability with current 193i lithography, which provides insights for process development.
- We propose a block re-distribution framework for SAB compliance and evaluate the impacts of SAB rules to the design flow.
- This is the first systematic co-optimization between SAB design rules and layout optimization techniques for N5 and beyond, which provides insights on future technology scaling.
- The experimental results on industrial 6-track standard-cell benchmarks demonstrate design flexibility of the SAB rules.

The rest of the this paper is organized as follows. Section II introduces basic backgrounds of SAB and the problem formulation. Section III explores the design rules for SAB under 193i lithography. Section IV describes the algorithms for SAB re-distribution. Section V validates the algorithms and Section VI concludes the paper.

II. Preliminaries

Without loss of generality, in this section, we use horizontal M2 layer under SAQP in the discussion.

A. SAB Patterns and Conflicts

Fig. 2 shows the SAQP process with alternating materials, where the 1st spacer is material A. After filling 2nd spacer, material B is filled instead of removing the 1st spacer in conventional SAQP. Then etching process with selectivity to material A, B, and 2nd spacer is necessary to go further. Details can be found in [4], [9].

Due to the selectivity, we can focus the study on one material, e.g., material A. The regularity of layout results in limited types of block patterns. The block patterns that may cause conflicts can be categorized into 4 types: *parallel*, *abutting*, *stacking* and *diagonal*, shown in Fig. 3. We separate the spacing rules required by different patterns with different symbols for easier illustration.

The conflicts from parallel patterns and abutting patterns can be resolved by increasing the horizontal distance or aligning two blocks. If the vertical distance of $2P$ is smaller than d_s for stacking patterns, they need to be transformed to diagonal patterns with enough corner-to-corner distance d_d. Similarly the conflicts from diagonal patterns can be resolved by moving blocks away from each other.

Line end extension is required to increase the distances where the amount of extension is restricted by nearest vias at two sides of a block. A via has minimum enclosure requirements for metal tips below and above the via, which is set to 0nm and 5nm. The maximum extension is constrained to 100nm to avoid large timing and power degradation.

B. Problem Formulation

To explore whether SAB is a viable solution for tight metal pitches in N7/N5, such as M2, we need to solve the following two problems.

Problem 1 (Design rule exploration)**.** *Given the lithography option of SAB and technology definitions in N7/N5, such as metal pitches and lithography capabilities, define design rules with enough space for design closure and meanwhile ensure manufacturability.*

Tight design rules with large lithography margin are likely to restrict the design space too much to achieve SAB-compliant layout, while

relaxed design rules may go beyond the printability of two 193i masks. Whether the design rules provide enough design space to resolve violations is critical for the validation of rules. Since current physical design flow does not support SAB yet, we attempt to approach SAB compliance in the post layout stage.

Problem 2 (Block optimization)**.** *Given a set of design rules and a design, optimize block distribution for design rule compliance, and meanwhile minimize the total cost of re-distribution such as the amount of line end extensions.*

The optimization procedure is mainly adopted to evaluate design rules. It might be difficult to reach zero violations with a post layout optimization, while the scale of remaining conflicts matters to the feasibility of early stage integration or additional ECO efforts.

III. SAB Design Rule Exploration

In this section, we explore design rules for SAB under 193i lithography and discuss the impacts to layout. The SAB rules are explored with the holistic consideration of typical DRC rules such as metal pitch, minimum metal area, single exposure spacing and metal tip-to-tip (implicitly considered by block width). Lithography retargeting and bias for optical proximity correction (OPC) are considered implicitly with the rules. As mentioned in previous section, the minimum required horizontal distance d_h for a diagonal pattern can be derived from d_d as follows,

$$d_h = \sqrt{\max(d_d^2 - 4P^2, 0)}. \quad (1)$$

Fig. 4(a) shows smaller d_d gives smaller d_h as the vertical distance is $2P$. The spacing rule for parallel patterns needs to satisfy both minimum area constraint A and single exposure spacing s,

$$d_p = \max(\frac{2A}{P}, s), \quad (2)$$

where we assume the metal width is $\frac{1}{2}P$. When the pitch scales down, minimum area A may dominate d_p instead of s.

A. Simple Rules

A straight-forward rule is to set a simple spacing value for all patterns in Fig. 3, i.e., $d_a = d_s = d_d = d_p$. This spacing rule is mainly limited by d_p from parallel patterns because it needs to satisfy both minimum area constraint of metal and single exposure spacing, while other patterns do not have the minimum area constraint. The simple rule is likely to forbid stacking patterns, since the end-to-end distance of $2P(P \leq 36)$ is too tight for 193i lithography.

B. Complex Rules

Tighten spacing rules. The simple rules are so strict that forbid all stacking patterns and require long distances for other patterns. While it is difficult to further push the edge-to-edge distance (d_p) for parallel patterns, other spacings such as d_a, d_s, d_d for abutting, stacking and diagonal patterns can be tighten to single exposure spacing s, since there is no constraint of minimum area for them.

Block end extension. However, this is not enough to resolve conflicts for stacking patterns. Considering the EPE margin of $\frac{3}{4}P$, we can slightly push back the block end to meet the spacing requirement. We define it as *block end extension* (bext) to differentiate from line end extension. The amount of block end extension is computed as,

$$bext = \max(\frac{1}{2}d_s - P, 0). \quad (3)$$

As shown in Fig. 4(b), the left EPE margin after the extension is,

$$mgn* = \frac{3}{4}P - bext = \begin{cases} \frac{7}{4}P - \frac{1}{2}d_s, & \text{if } d_s \geq 2P, \\ \frac{3}{4}P, & \text{otherwise.} \end{cases} \quad (4)$$

In this way, conflicts from stacking patterns can be resolved by either block end extension or transforming to diagonal patterns by line end extension. At the same time, diagonal patterns also obtain another option to resolve conflicts, i.e., converting to stacking patterns by line end extension and performing block end extension.

The block end extension is only enabled for stacking patterns, though it also helps resolve conflicts in diagonal patterns. The reason lies in the support for aligned blocks which requires the lithographic system to be optimized for elongated block constructs, e.g., the illumination may have to go dipole-like sources, where we are losing EPE control over the block end. Therefore, it is better to keep as much EPE margin as possible and it is not encouraged to enable block end extension for diagonal patterns.

Table I shows the impact of these options with various pitches and single exposure spacing s by comparing conventional blocks with SAB. We set $d_a = d_d = d_s$ in the computation since none of them subject to minimum area constraint of lines. Thus only d_s is shown in the table. The column "mgn" denotes the EPE margin for block end without block end extension. The column "bext" denotes the amount of block end extension required to resolve conflicts, while "mgn*" denotes the EPE margin for a block after block end extension, which is the worst case of EPE margin, since not all blocks need block end extension. Highlighted entries denote that there is enough margin for blocks to tolerate variations. As both simple and complex rules of SAB have the same d_p, only one column is shown. The third row from top shows how each entry is computed. The minimum area constraint A is set to 1452nm^2 [10]. We assume the spacing of single exposure optical lithography for blocks is around 80nm [2], [11], [12]. If we use the same configurations as [4], e.g., overlay error of 1nm, block end variation of 2nm, line critical dimension (CD) variation of 1nm, the required EPE margin can be estimated as $3\sigma \approx$ 8nm.

Even for 32nm and 36nm pitches, conventional blocks only have marginal values to tolerate process variation, while SAB has much larger EPE margins. If we do not perform block end extension, like that in simple rules, then stacking patterns have to be forbidden as vertical distance between two blocks of $2P < d_s$. By allowing block end extension, we enable stacking patterns at the cost of losing EPE margins. For 32nm and 36nm pitches, the margins are large enough for block end extension with various lithography spacings, while d_s need to be tighten to 68nm for enough margin in 24nm pitch, which goes beyond the limitation of 193i lithography and probably necessitates EUV or SAB with four 193i masks. Considering that 24nm is likely to be the M2 pitch for industry N3 [12], they might be options.

IV. SAB Optimization Algorithms

A. Overall Flow

Fig. 5 gives the overall flow of SAB optimization. The framework takes input layout and process lines, vias and blocks for each material separately. We construct potential conflict graph considering the movable bounds of blocks [8] and compute independent components to reduce problem sizes. Due to tight pitches, each independent component may still be large, so metis based min-cut partitioning is performed ro remove some edges by reducing the movable bounds of some blocks [13]. The kernel optimization for each component includes generation, pruning and graph construction of candidate movements followed by candidate assignment. After kernel optimization, the majority of conflicts has been removed. We incrementally fix small regions with conflicts to further refine the solutions iteratively where more candidate movements are generated for larger solution spaces. Block end extension is performed for stacking patterns before output.

B. Candidate Movement Assignment

Given the candidate movements for each block, we can construct conflict graph $G = (V, E)$ for candidate movements, where each candidate movement is a vertex in vertex set V and two movements are connected with an edge in E if they result in a conflict or they belong to the same block. The target of the candidate movement assignment is to select one candidate movement for each block such that the final conflicts and total cost are minimized.

The full ILP formulation is summarized as follows,

$$\min \quad \sum w_{ij} \cdot x_{ij} + \sum w_e \cdot t_e, \tag{5a}$$

$$\text{s.t.} \quad \sum_j x_{ij} = 1, \tag{5b}$$

$$x_{i_1 j_1} + x_{i_2 j_2} \leq t_e + 1, \forall e \in E \backslash E_{as}, \tag{5c}$$

$$x_{i_1 j_1} + x_{i_3 j_3} - x_{i_2 j_2} \leq t_e + 1, \forall e \in E_{as}, \tag{5d}$$

$$x_{ij} \in \{0, 1\}, t_e \geq 0, \tag{5e}$$

where x_{ij} is a binary variable denoting whether the jth candidate movement of block b_i is selected, and w_{ij} is its cost. Variable t_e denotes an edge in the conflict graph and whether both of its vertices are selected. As t_e is constrained by x_{ij}, there is no need to constrain it to integers; i.e., it is a continuous variable. Weight w_e is the weight of edge e. We use E_{as} to denote the set of conflict edges from *aligned stacking patterns* which the details will be explained later. Eq. (5c) and Eq. (5d) are the *conflict constraints* for two candidate movements of different blocks. Eq. (5b) guarantees that only one candidate movement will be selected for one block. The objective in Eq. (5a) minimizes total number of conflicts and total cost of candidate movements selected.

Edges for conflicted candidate movements are assigned with a large weight M, while additional edges with weight w_s are inserted if two candidate movements forms stacking patterns. Depending on different pitches and rules, stacking patterns may become a conflict or not, so a generic way is to integrate them into edge weights of the graph. If they are regarded as a conflict, we set $w_s = M$; otherwise, w_s is assigned with a small cost, e.g., half the maximum line end extension in the experiment, to minimize block end extension. For simplicity, in this section, we use the phrase *conflict* to generalize real conflicts and stacking patterns if not specially mentioned, since the latter is incorporated into conflict edges. Therefore, the objective in Eq. (5a) includes costs from conflicts and stacking patterns.

The set E_{as} of *aligned stacking patterns* is introduced for the multiple aligned blocks, where Fig. 6(b) gives three candidate movements vertically aligned. Supposing their corresponding variables are $x_{i_1 j_1}$, $x_{i_2 j_2}$, and $x_{i_3 j_3}$, respectively, now consider the conflict constraint between c_1 and c_3. If c_2 is selected, there is no conflict between c_1 and c_3; otherwise, they generate a conflict. Eq. (5c) cannot properly handle such conditions, so we constrain it with *aligned conflict constraints* as Eq. (5d). If $x_{i_2 j_2}$ is 0, Eq. (5d) becomes the same format as Eq. (5c); otherwise, it does not constrain anything since the left hand side is always no larger than 2. Eq. (5d) can be extended to handle more than 3 aligned blocks in Fig. 6(c) as well.

V. Experimental Results

Our framework is implemented in C++ and tested under Linux server with 2.6GHz Xeon CPU and 48GB RAM. Industrial 6-track standard-cell designs with 32nm M2 pitch for double patterning blocks are used in the experiments. The open-source *lpsolve* is used as the ILP solver [14]. Table II shows the comparison between different rules for various benchmarks with 70% to 80% utilization of cells. Column "#blks" denotes the number of blocks. Column "#icn" denotes initial number of conflicts between blocks and column "#cn" denotes number

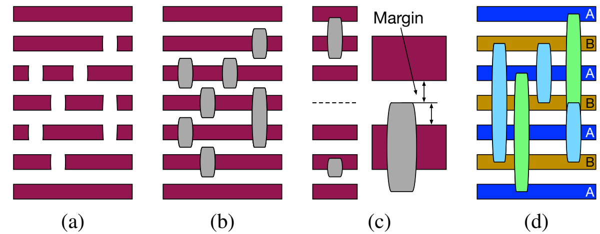

Fig. 1: Example of (a) horizontal M2 lines, (b) lines with conventional blocks, (c) block end variations, (d) SAB implementation of lines with light blue blocks for material A and light green ones for material B.

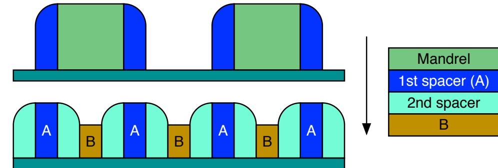

Fig. 2: SAQP process for alternating materials.

of conflicts after SAB optimization. Column "ext" denotes the average length of line end extension in nanometer for each block. Column "bext" indicates the percentage of block end extension to all block ends. Column "T" shows the runtime in seconds.

We compare the results of simple rules and complex rules using the configurations of $(P = 32, s = 80)$ and $(P = 32, s = 84)$ in Table I. Since the benchmarks are not designed for SAB, there are considerable conflicts at the beginning. Our framework is able to reduce conflicts effectively, while the number of final conflicts varies from different rules. Simple rules end up 22x more final conflicts, 3x length of average line end extension and 4x runtime compared with complex rules, while complex rules drop the amount of average final conflicts to around 100, which is much more applicable to be fixed by either ECO effort or early planning in placement and routing. If we relax lithography spacing s to 84nm, then there are 11% more conflicts with 7% longer average line end extension and 5% more block end extension. We also observe that the average length of line end extension is only around 5-10nm, which means most conflicts are fixed by local adjustment. Although the average amount of block end extension is only around 12%, it removes conflicts for stacking patterns effectively. In simple rules, stacking patterns are actually forbidden due to $2P < d_s$ (see Section III). Therefore, enabling stacking patterns is critical for design closure in advanced nodes.

VI. Conclusion

In this paper, we propose the first study on design technology co-optimization for SAB in N5 and beyond. We propose different design rules that scale together with pitches and explore the design space with a post optimization framework, which provides the insights for the direction of process advancement. The framework is flexible to various design constraints for SAB. Future work includes the optimization in early design stages for SAB friendliness.

Acknowledgment

This work is supported in part by NSF and the University Continuing Fellowship from The University of Texas at Austin.

References

[1] P. Raghavan, F. Firouzi, L. Matti, P. Debacker, R. Baert, S. Sherazi, D. Trivkovic, V. Gerousis, M. Dusa, J. Ryckaert, *et al.*, "Metal stack optimization for low-power and high-density for n7-n5," in *SPIE Advanced Lithography*, pp. 97810Q–97810Q, International Society for Optics and Photonics, 2016.

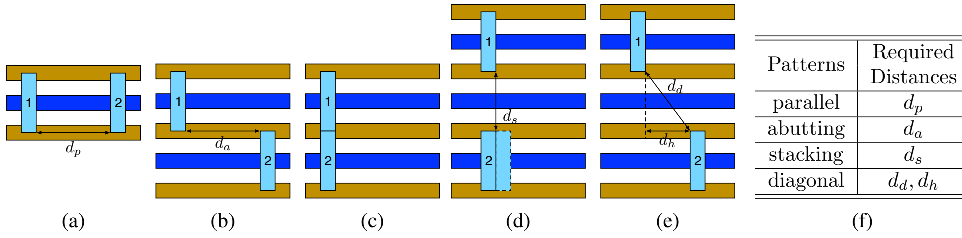

Fig. 3: Various configurations between block 1 and block 2: (a) parallel pattern, (b) abutting pattern, (c) vertically aligned pattern which can be merged, (d) stacking patterns, (e) diagonal pattern. (f) Summary of minimum required distances.

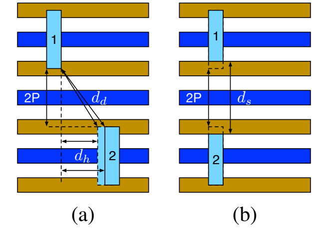

Fig. 4: Example of (a) diagonal pattern with complex rules and (b) stacking patterns with complex rules.

TABLE I: Impacts of design rules (nm)

		block		SAB simple rules			SAB complex rules			
P	s	mgn	d_p	d_s	d_h	mgn	d_s	d_h	bext	mgn*
		$\frac{1}{4}P$	Eq. (2)	d_p	Eq. (1)	$\frac{3}{4}P$	s	Eq. (1)	Eq. (3)	Eq. (4)
24	84	6	121	121	112	**18**	84	69	18	0
	80		121	121	112	**18**	80	64	16	2
	68		121	121	112	**18**	68	49	10	**8**
28	84	7	104	104	88	**21**	84	63	14	7
	80		104	104	88	**21**	80	58	12	**9**
32	84	**8**	91	91	65	**24**	84	55	10	**14**
	80		91	91	65	**24**	80	48	8	**16**
36	84	**9**	84	84	44	**27**	84	44	6	**21**
	80		81	81	38	**27**	80	35	4	**23**

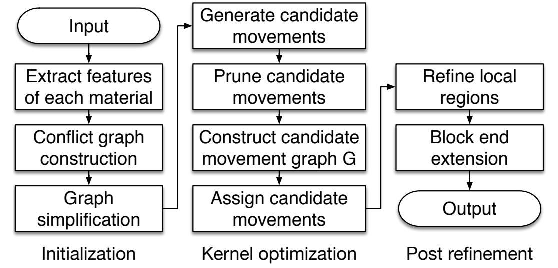

Fig. 5: Overall flow of SAB optimization.

TABLE II: Comparison of different rules for $(P = 32, s = 80)$ and $(P = 32, s = 84)$

Design	Util	#blks	Simple rules $(P = 32, s = 80)$					Complex rules $(P = 32, s = 80)$					Complex rules $(P = 32, s = 84)$				
			#icn	#cn	ext	bext	T(s)	#icn	#cn	ext	bext	T(s)	#icn	#cn	ext	bext	T(s)
b1	80%	3K	1133	106	22.6	0.0%	5.8	982	6	7.7	11.9%	3.9	1006	6	8.0	12.7%	5.6
b2	70%	14K	4410	288	17.9	0.0%	17.3	3883	11	5.5	10.3%	5.0	4028	11	6.0	10.8%	4.1
b3	80%	19K	6760	845	23.7	0.0%	109.7	5895	46	8.2	13.0%	27.0	6151	51	8.8	13.5%	17.4
b4	70%	46K	13758	948	16.9	0.0%	42.8	12081	54	5.6	9.0%	14.0	12470	57	6.1	9.5%	20.3
b5	70%	56K	26633	1782	27.2	0.0%	99.8	23459	74	8.9	14.7%	44.9	24224	84	9.6	15.4%	56.7
b6	70%	211K	66583	4633	18.1	0.0%	256.6	58679	204	5.9	9.9%	100.0	60609	226	6.4	10.4%	114.6
b7	80%	218K	72768	8285	20.5	0.0%	1077.5	63256	354	7.3	11.6%	166.3	65830	396	7.9	12.2%	173.7
avg.	-	-	27435	2412	21.0	0.0%	229.9	24034	107	7.0	11.5%	51.6	24903	119	7.5	12.1%	56.1
ratio	-	-	1.14	22.55	2.99	0.00	4.46	1.00	1.00	1.00	1.00	1.00	1.04	1.11	1.07	1.05	1.09

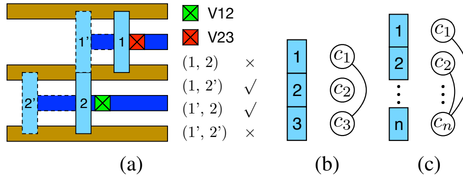

Fig. 6: (a) Example of candidate movements for block 1 and block 2 where combinations like either (1, 2') or (1', 2) can resolve the conflict. Example of (b) 3 vertically aligned candidate movements for 3 blocks and (c) n vertically aligned candidate movements for n blocks.

[2] L. Liebmann, A. Chu, and P. Gutwin, "The daunting complexity of scaling to 7nm without EUV: Pushing DTCO to the extreme," in *Proc. SPIE*, vol. 9427, 2015.

[3] W. Gillijns, S. Sherazi, D. Trivkovic, B. Chava, B. Vandewalle, V. Gerousis, P. Raghavan, J. Ryckaert, K. Mercha, D. Verkest, *et al.*, "Impact of a SADP flow on the design and process for N10/N7 metal layers," in *Proc. SPIE*, vol. 9427, 2015.

[4] T. Han, H. Liu, and Y. Chen, "A paradigm shift in patterning foundation from frequency multiplication to edge-placement accuracy: a novel processing solution by selective etching and alternating-material self-aligned multiple patterning," in *SPIE Advanced Lithography*, pp. 977718–977718, International Society for Optics and Photonics, 2016.

[5] Y. Du, H. Zhang, M. D. Wong, and K.-Y. Chao, "Hybrid lithography optimization with e-beam and immersion processes for 16nm 1d gridded design," in *17th Asia and South Pacific Design Automation Conference*, pp. 707–712, IEEE, 2012.

[6] Y. Ding, C. Chu, and W.-K. Mak, "Throughput optimization for sadp and e-beam based manufacturing of 1d layout," in *Proceedings of the 51st Annual Design Automation Conference*, pp. 1–6, ACM, 2014.

[7] K. Han, A. B. Kahng, H. Lee, and L. Wang, "Ilp-based co-optimization of cut mask layout, dummy fill, and timing for sub-14nm beol technology," in *SPIE Photomask Technology*, pp. 96350E–96350E, International Society for Optics and Photonics, 2015.

[8] J. Kuang, E. F. Young, and B. Yu, "Incorporating cut redistribution with mask assignment to enable 1d gridded design," in *Proceedings of the 35th International Conference on Computer-Aided Design*, p. 48, ACM, 2016.

[9] H. Liu, T. Han, J. Zhou, and Y. Chen, "Layout decomposition and synthesis for a modular technology to solve the edge-placement challenges by combining selective etching, direct stitching, and alternating-material self-aligned multiple patterning processes," in *SPIE Advanced Lithography*, pp. 97810P–97810P, International Society for Optics and Photonics, 2016.

[10] B. Chava, D. Rio, Y. Sherazi, D. Trivkovic, W. Gillijns, P. Debacker, P. Raghavan, A. Elsaid, M. Dusa, A. Mercha, *et al.*, "Standard cell design in n7: Euv vs. immersion," in *SPIE Advanced Lithography*, pp. 94270E–94270E, International Society for Optics and Photonics, 2015.

[11] A. Mallik, J. Ryckaert, A. Mercha, D. Verkest, K. Ronse, and A. Thean, "Maintaining Moore's law - enabling cost-friendly dimensional scaling," in *Proc. SPIE*, vol. 9422, 2015.

[12] L. Liebmann, J. Zeng, X. Zhu, L. Yuan, G. Bouche, and J. Kye, "Overcoming scaling barriers through design technology cooptimization," in *VLSI Technology, 2016 IEEE Symposium on*, pp. 1–2, IEEE, 2016.

[13] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM Journal on scientific Computing*, vol. 20, no. 1, pp. 359–392, 1998.

[14] LPSOLVE, "lp_solve, a Mixed Integer Linear Programming solver." http://lpsolve.sourceforge.net/5.5, 2016.