

Layout Decomposition Co-optimization for Hybrid E-Beam and Multiple Patterning Lithography

Yunfeng Yang, Wai-Shing Luk, David Z. Pan, *Fellow, IEEE*, Hai Zhou, *Senior Member, IEEE*, Changhao Yan, Dian Zhou, *Senior Member, IEEE*, and Xuan Zeng, *Member, IEEE*

Abstract—As the feature size keeps scaling down and the circuit complexity increases rapidly, a more advanced hybrid lithography, which combines multiple patterning and e-beam lithography (EBL), is promising to further enhance the pattern resolution. In this paper, we formulate the layout decomposition problem for this hybrid lithography as a minimum vertex deletion K -partition problem, where K is the number of masks in multiple patterning. Stitch minimization and EBL throughput are considered uniformly by adding a virtual vertex between two feature vertices for each stitch candidate during the conflict graph construction phase. For $K = 2$, we propose a primal-dual (PD) method for solving the underlying minimum odd-cycle cover problem efficiently. In addition, a chain decomposition algorithm is employed for removing all “non-cyclable” edges. Furthermore, we investigate two versions of the PD method, one with planarization and one without. For $K > 2$, we propose a random-initialized local search method that iteratively applies the primal-dual solver. Experimental results show that compared with a two-stage method, our proposed methods reduce the EBL usage by 65.5% with double patterning and 38.7% with triple patterning on average for the benchmarks.

Index Terms—Hybrid lithography, e-beam, multiple patterning, layout decomposition, primal-dual, graph bipartization.

I. INTRODUCTION

AS the resolution limit of conventional optical lithography is not capable to pattern sub-20nm half-pitch for the semiconductor industry, several next generation lithography methods, such as *extreme ultra-violet* (EUV) lithography, *electron beam lithography* (EBL), and *multiple patterning lithography* (MPL) have been proposed. However, EUV and

EBL are not yet available for volume production. Although stencil planning [2] is studied to improve EBL throughput, it might not suffice for very large layouts. On the other hand, MPL like *double patterning lithography* (DPL) [3]–[5] and *triple patterning lithography* (TPL) [6]–[9] can significantly enhance the capability of the conventional 193nm lithography technology. However, many unresolvable conflicts still cannot be eliminated even by slicing the features in the layout into smaller pieces, especially for very complex layouts.

In order to further eliminate the remaining conflicts for MPL, some modification technologies have been proposed recently, such as layout compaction [10] and post-routing layer assignment [11]. However, they inevitably change the layouts more or less, which may degrade the electrical characteristics.

Single lithography technology may not be sufficient for producing chips with decreasing feature size and increasing complexity. Thus, in the past decade, industry and academia have already explored the combination of different lithography technologies, especially the combination of optical lithography and EBL [12] [13]. The hybrid lithography of optical lithography and EBL goes through two main stages: (1) the high throughput but low resolution optical exposure which manufactures the majority of features in the layout; (2) the high resolution but low throughput e-beam exposure which produces the features with extremely tight spacing. If MPL is adopted in the first stage, the patterning capability of the first stage can be further enhanced, which can reduce the EBL utilization in the second stage.

This combination of high throughput optical lithography and high resolution e-beam lithography can lead to a more powerful patterning capability. Recent studies [14]–[16] show that the results are promising. Throughput optimization for *self-aligned double patterning* (SADP) and e-beam based manufacturing of 1D layout was investigated in [13] [14]. In [15], Gao et al. introduced a method for the SADP layout decomposition with complementary EBL. However, since stitch insertion is not allowed in the self-aligned process, large EBL utilization is necessary to resolve the conflicts. In [16], Tian et al. presented the hybrid lithography of LELELE triple patterning lithography and EBL for standard-cell based row structure layouts. Nevertheless, their method is only for standard-cell based row structure designs and stitch insertion is not considered, which may incur excessive EBL utilization.

The idea for the layout decomposition co-optimization for the hybrid lithography of MPL and EBL is illustrated in Fig. 1.

Manuscript received January 27, 2015; revised May 18, 2015, August 13, 2015 and October 29, 2015; accepted December 10, 2015. Date of current version December 15, 2015. This work was supported partly by the NSFC F040204, 61125401, 61376040, 61228401, and 61274032, partly by NSF under CCF-1115550, CCF-1218906 and CNS-1441695, partly by National Basic Research Program of China under 2011CB309701, partly by National Major Science and Technology Special Project of China 2011ZX01035-001-001-003 and 2014ZX02301002-002, partly by Shanghai Science and Technology Committee 13XD1401100, and China Thousand Talent Plan Program Grant. This paper was recommended by Associate Editor Zhuo Li.

Y. F. Yang, W. S. Luk, C. H. Yan, and X. Zeng are with the State Key Laboratory of ASIC and System, Microelectronics Department, Fudan University, Shanghai 201203, China (e-mail: xzeng@fudan.edu.cn).

D. Z. Pan is with the Department of Electrical and Computer Engineering, University of Texas at Austin, USA.

H. Zhou is with the Department of Electrical Engineering and Computer Science, Northwestern University, USA.

D. Zhou is with the Department of Electrical Engineering, University of Texas at Dallas, USA.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier XXXX

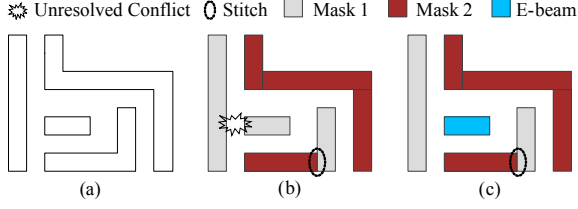


Fig. 1. Hybrid lithography of double patterning and e-beam. (a) The original layout. (b) The double patterning decomposition result with an unresolvable conflict and a stitch. (c) The co-optimization result of double patterning decomposition and e-beam lithography utilization.

As shown in Fig. 1(b), after the double patterning lithography layout decomposition, there is still an unresolvable conflict, which cannot be eliminated by any DPL decomposition method. To manufacture the conflicting features, EBL is adopted as shown in Fig. 1(c). However, EBL utilization should be minimized since EBL throughput is very low compared with the optical lithography.

In this work, we solve the layout decomposition co-optimization problem for general layouts, which can enable the hybrid lithography combining LELE-style MPL and EBL. First, we formulate the problem as a kind of *minimum vertex deletion K-partition problem*, where K refers to the number of masks in multiple patterning. Stitch minimization and EBL throughput are considered simultaneously by adding a virtual vertex for each stitch candidate during the conflict graph construction phase.

We first consider the problem for $K = 2$. The minimum vertex deletion K -partition problem then reduces to the *minimum vertex deletion graph bipartization problem*. Recall that a graph is bipartite if and only if it does not contain any odd cycle. Thus, the problem is equivalent to the *minimum odd-cycle cover* problem. In this paper, we present two versions of the *primal-dual* (PD) method for solving this problem, one with planarization and one without. Observing that the underlying conflict graph is nearly planar, we investigate a version of PD method based on [17]. Planarization procedure is invoked in this method. We also investigate another version of the PD method which does not require any planarization process.

In order to correctly compute the dual variables, the primal-dual method requires removing all “non-cyclable” edges, i.e., edges which cannot be part of any cycle in the graph. They include all *dangling* edges and *bridge* edges. Recently, Schmidt presented a surprisingly simple chain decomposition algorithm for testing on 2-vertex- and 2-edge-connectivity of a graph in linear time [18]. Interestingly, we find that the algorithm can also be used for finding all “non-cyclable” edges as well.

Nevertheless, above techniques unfortunately cannot directly be extended for $K > 2$. In such cases, we propose a random-initialized local search method that iteratively applies the primal-dual solver for a sequence of graph bipartization subproblems. The framework can run in parallel easily.

Besides, we also present a method that is based on the *integer linear programming* (ILP) formulation for the D-PL+EBL decomposition. The method is used as a baseline for comparing the proposed methods.

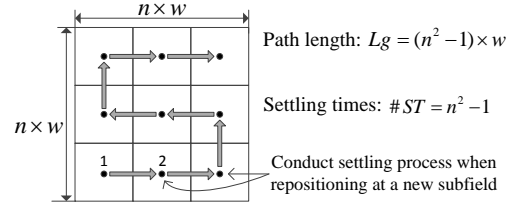


Fig. 2. Path length and stage settling times for the sequential writing path. We assume the head starts at the first subfield (the left-bottom one). The layout is divided into $n \times n$ subfields. The size of each subfield is $w \times w$.

Moreover, we review the overall EBL writing time since it is the bottleneck of the throughput improvement. We discuss how possible it is to reduce the stage movement and settling time on our system.

The rest of the paper is organized as follows. We review the EBL system in Sec. II. The problem formulation is introduced in Sec. III. A two-stage method is introduced in Sec. IV. Two versions of the primal-dual method for DPL and EBL, and the random-initialized local search method for MPL and EBL are presented in Sec. V. Sec. VI shows the experimental results, followed by a conclusion in Sec. VII.

II. REVIEW OF THE EBL SYSTEM

First, we review the EBL system. In a typical EBL system, the features are written by beam deflection and stage movement [19]. Since the beam deflection range is limited, the layout is divided into smaller subregions named *subfields*. This way enables the e-beam head to cover all the features within a subfield. After producing the features within a subfield, the e-beam head moves to another subfield by stage movement. Besides, when the head arrives at a new subfield, a minimum time called *stage settling time* is needed to do repositioning. Thus, the total writing time consists of three major parts: (1) the exposure time T_e for producing features in the layout, (2) the stage movement time T_m when the head moves between different subfields, and (3) the stage settling time T_s when the head repositions itself. Namely, the total time is

$$t_{all} = T_e + T_m + T_s. \quad (1)$$

To reduce the exposure time, MPL is adopted to produce the majority of features in the layout and the layout decomposition methods will be introduced in the following sections. However, how to reduce the movement time and settling time depends on one’s own experience. Here, we can give some suggestions. In many EBL systems, the stage settling time is often much larger than the stage movement time. If the number of subfields is reduced, the settling time can also be reduced. Nevertheless, since the subfield size is limited by the system working range, pattern resolution and digital-to-analog converter bits, the tradeoff between the subfield size and the limiting factors is usually made case by case.

Then, we show the EBL writing time for some designs. We assume using the EBL system JEOL6300FS. The related data are in Table I. We use UVIII as the resist ($100\mu\text{C}/\text{cm}^2$). The feature exposure time is calculated by the commercial software BEAMER. We use the sequential path in Fig. 2. After

TABLE II
RESULTS OF THE OVERALL E-BEAM WRITING TIME.

Cases	Feature exposure time		Stage movement time				Stage settling time		Total time
	Area($\mu\text{m} \times \mu\text{m}$)	$T_e(\text{ms})$	Subfield Size($\mu\text{m} \times \mu\text{m}$)	#SF	$Lg(\mu\text{m})$	$T_m(\text{ms})$	#ST	$T_s(\text{ms})$	
S38417	35.87	72.0	62.5 \times 62.5	2 \times 2	187.5	6.25	3	549.9	628.15
			120 \times 120	1 \times 1	0.0	0.00	0	0.0	72.00
S35932	98.20	198.0	62.5 \times 62.5	2 \times 2	187.5	6.25	3	549.9	754.15
			120 \times 120	1 \times 1	0.0	0.00	0	0.0	198.00
S38584	94.80	192.0	62.5 \times 62.5	2 \times 2	187.5	6.25	3	549.9	748.15
			120 \times 120	1 \times 1	0.0	0.00	0	0.0	192.00
S15850	97.44	192.0	62.5 \times 62.5	2 \times 2	187.5	6.25	3	549.9	748.15
			120 \times 120	1 \times 1	0.0	0.00	0	0.0	192.00
Cluster	11882.20	23958.0	62.5 \times 62.5	20 \times 20	24937.5	831.25	399	73136.7	97925.95
			110 \times 110	11 \times 11	13200.0	440.00	120	21996.0	46394.00
			1000 \times 1000	2 \times 2	3000.0	100.00	3	549.9	24607.90

TABLE I
JEOL6300FS DATA SHEET AND E-BEAM RESIST

Stage settling time (ms)	183.3	
Stage moving speed (mm/s)	30	
Subfield size($\mu\text{m} \times \mu\text{m}$)	Min	62.5 \times 62.5
	Max	1000 \times 1000
Optimized exposure dose ($\mu\text{C}/\text{cm}^2$)	1500	
Beam current (pA)	500	
Accelerating voltage (keV)	100	
Minimum beam spot size (nm)	3	
Stage precision (nm)	0.6	
Digital-to-analog converter bits	20	
E-beam resist	PMMA	1500
	UVIII	30-100

DPL+EBL decomposition, we extract the features for EBL. The results are in Table II. We denote “Area” the exposure area, #SF the number of subfields, Lg the writing path length, and #ST the settling times. As shown in Table II, for the smaller layouts like S38417, if we use a smaller subfield (62.5 $\mu\text{m} \times 62.5\mu\text{m}$), the exposure time is only 72ms, while the stage settling time is 549.9ms. However, if we use a larger subfield (120 $\mu\text{m} \times 120\mu\text{m}$), we need not perform the stage movement and stage settling. Then, the total writing time can be reduced by 88.5%. We can get similar results for the other cases. Thus, we conclude that maximizing the subfield size can help improve the throughput on our system. Nevertheless, the subfield size is restricted by the limiting factors.

III. PROBLEM FORMULATION

A. Preliminaries

The hybrid lithography of MPL and EBL consists of two stages. In the first stage, high throughput multiple patterning is conducted to resolve as many as possible conflicts. At the same time, stitches are inserted when necessary. Since stitches will increase manufacturing cost, they should be minimized. In the second stage, high resolution e-beam exposure is operated to eliminate the unresolvable conflicts of the first stage. Since the throughput of e-beam is very low, the utilization of e-beam should be minimized to reduce the writing time. Moreover, since producing a feature by two different lithography technologies will induce greater manufacturing cost, we enforce that each feature be produced by only one lithography technology, i.e., either by MPL or EBL.

We adopt the conventional *variable-shaped rectangular beams* (VSB), as in [15] [16]. However, unlike [15] [16], we will minimize the total area of VSB rather than the cardinality of VSB. Strictly speaking, the writing time of VSB is determined by the total area of VSB [20], whereas it does not affect the stage movement and settling time as described in Sec. II. Thus, our formulation is more realistic in practice.

Our conflict graph construction is based on [3], [21]. Given a layout which is specified by polygon features, the features are first fractured into rectangles. Note that the rectangles may be further sliced into smaller pieces. Then, a conflict graph $G = (V, E)$ is constructed according to a given minimum coloring distance d_{MP} for MPL, where a node $v \in V$ denotes a rectangle and an edge $e \in E$ denotes a conflict or stitch candidate.

B. Mathematical Formulation

Notations. Here, we introduce some notations throughout this paper.

- E_c , the set of conflict edges,
- E_s , the set of stitch candidates,
- K , the number of masks for MPL steps,
- d_{DP} , the minimum coloring distance for DPL,
- d_{TP} , the minimum coloring distance for TPL.

Accordingly, the co-optimization of MPL decomposition and EBL utilization is formulated as follows:

Given: A graph $G = (V, E_c \cup E_s)$ and a weight function $A : V \mapsto \mathbb{R}$, which is typically set to the area of the corresponding rectangle of node v .

Find: A subset $V' \subseteq V$ such that the subgraph induced by V' is K -colorable, and the corresponding color assignment $c : V' \mapsto [1..K]$. In addition, if $v \in V \setminus V'$, then for all $(u, v) \in E_s$, u must also be in $V \setminus V'$ because v and u belong to the same polygon.

Minimize: The weighted sum of the EBL throughput cost and the MPL stitches, i.e.,

$$\min \sum_{v \in V \setminus V'} \alpha A_v + \sum_{(u,v) \in E'_s} \beta, \quad (2)$$

where α, β are given weighting factors, and $E'_s = \{(u, v) \mid (u, v) \in E_s, u \in V', v \in V', c_u \neq c_v\}$, denoting the set of used stitch candidate edges. Note that $\alpha \gg \beta$ usually, since the manufacturing cost for EBL is much higher than that of stitches.

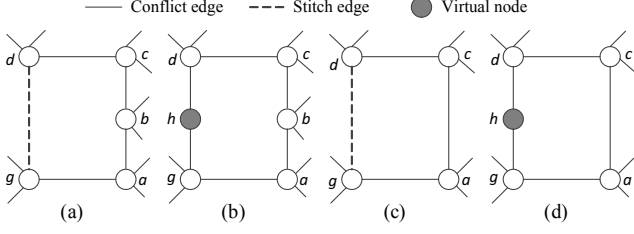


Fig. 3. Substitute a stitch edge with a virtual node and two virtual edges. (a) a, b, c, d and g form an even cycle where a stitch edge is not counted. (b) Substitute the stitch edge (g, d) with (h, d) and (h, g) . a, b, c, d, h and g is still an even cycle. (c) a, c, d and g form an odd cycle where a stitch edge is not counted. (d) Substitute the stitch edge (g, d) with (h, d) and (h, g) . a, c, d, h and g is still an odd cycle.

C. Handling Stitch Edges by Vertex Deletion

In order to consider the stitch minimization and EBL throughout uniformly by vertex deletion, we substitute each stitch edge with one virtual node and two corresponding virtual edges. Note that this transformation does not change the bipartite property of the original graph. The weight of a virtual node is usually set to the weight of the corresponding stitch. The degree of a virtual node is always two. An example is shown in Fig. 3.

The virtual node is closely related with the candidate stitch, which is described by the following theorem:

Theorem 1. *In the DPL+EBL decomposition problem ($K = 2$), if a virtual node is deleted only when its existence leads to an odd cycle, then retaining a virtual node is equivalent to making the corresponding stitch candidate unused, while deleting a virtual node is equivalent to inserting a stitch, i.e., making the corresponding stitch candidate used.*

Proof. Let v_0 be a virtual node, v_1 and v_2 be the two corresponding neighbor vertices of the virtual edges. Note that v_1 and v_2 are from the same polygon.

(1) For the case that a virtual node is retained, if both v_1 and v_2 are retained, then a feasible bi-coloring solution satisfies,

$$\begin{cases} \text{color}(v_1) \neq \text{color}(v_0), \\ \text{color}(v_2) \neq \text{color}(v_0), \end{cases} \Rightarrow \text{color}(v_1) = \text{color}(v_2).$$

Thus, the candidate stitch is unused. If either v_1 or v_2 is deleted, then both v_1 and v_2 must be deleted, because v_1 and v_2 belong to the same polygon which should be manufactured by only one lithography technology, i.e., EBL.

(2) For the case that a virtual node is deleted, then there is an odd cycle C which contains v_0, v_1 and v_2 . Let the path $PH = C \setminus \{(v_1, v_0) \cup (v_0, v_2)\}$. Starting from v_1 , for any node v on PH , the distance $\text{dist}(v_1, v)$ between v_1 and v (i.e., the number of edges between v_1 and v on path PH) is,

$$\begin{cases} \text{color}(v_1) \neq \text{color}(v), & \text{if } \text{dist}(v_1, v) \text{ is odd,} \\ \text{color}(v_1) = \text{color}(v), & \text{if } \text{dist}(v_1, v) \text{ is even.} \end{cases}$$

Since $|C|$ is odd, thus $|PH| = |C \setminus \{(v_1, v_0) \cup (v_0, v_2)\}|$ is also odd. We can obtain that $\text{color}(v_1) \neq \text{color}(v_2)$, because $\text{dist}(v_1, v_2)$ is odd. Therefore, the candidate stitch is used. \square

The co-optimization problem of MPL decomposition and EBL utilization can be reformulated as a kind of minimum

weight vertex deletion K -partition problem described in [22], which is known to be NP-hard in general. When $K = 2$, it reduces to a *minimum vertex deletion bipartization* (MVDB) problem. Note that the pure MPL layout decomposition is a minimum (weight) edge deletion K -partition problem. Thus, DPL decomposition is a minimum edge deletion bipartization problem. Therefore, the pure MPL decomposition and the MPL+EBL decomposition are two different problems in nature. Moreover, MVDB problem is more difficult than the minimum edge deletion bipartization counterpart in the sense that the minimum edge deletion bipartization problem can be solved optimally in polynomial time for planar graphs [23], while to obtain the optimal solution for MVDB problem is still NP-hard for a planar graph G if $\Delta(G) \geq 4$, where $\Delta(G)$ is the maximum degree of the nodes in G [24].

For $K > 2$, we propose a method that applies the MVDB solver iteratively, and thus the vertex deletion K -partition problem is reduced to a sequence of MVDB problems.

D. ILP Formulation for DPL+EBL Decomposition

In this section, we introduce an ILP formulation for DPL+EBL decomposition. Given the set of conflict edges E_c and the set of stitch candidate edges E_s , the ILP formulation for DPL+EBL decomposition is derived as follows.

Variables used in the ILP formulation:

- $b_v = \begin{cases} 1, & \text{node } v \text{ is assigned to the first mask,} \\ 0, & \text{node } v \text{ is assigned to the second mask,} \end{cases}$
- $x_v = \begin{cases} 1, & \text{node } v \text{ is assigned to EBL "mask",} \\ 0, & \text{otherwise,} \end{cases}$
- $s_{uv} = \begin{cases} 1, & \text{stitch candidate between } u \text{ and } v \text{ is used,} \\ 0, & \text{otherwise.} \end{cases}$

An objective function that needs to be minimized is the weighted sum of the total cost of e-beam usage and the number of stitches. Let α be the weight of e-beam and β be the weight of stitch edges.

The objective function used in the ILP formulation:

Minimize $(\sum_{v \in V} \alpha \cdot A_v \cdot x_v + \sum_{(u,v) \in E_s} \beta \cdot s_{uv})$.

Constraints used in the ILP formulation:

$$b_u + b_v + x_u + x_v \geq 1, \quad \forall (u, v) \in E_c. \quad (3)$$

$$b_u + b_v - x_u - x_v \leq 1, \quad \forall (u, v) \in E_c. \quad (4)$$

$$x_u - x_v = 0, \quad \forall (u, v) \in E_s. \quad (5)$$

$$b_u - b_v \leq s_{uv}, \quad \forall (u, v) \in E_s. \quad (6)$$

$$-b_u + b_v \leq s_{uv}, \quad \forall (u, v) \in E_s. \quad (7)$$

Constraints (3) and (4) specify that a conflict edge is resolved by either assigning the two vertices to different masks of DPL or manufacturing one of the vertices by EBL. Constraint (5) is used for making the rectangles of a polygon produced by only one lithography (i.e. DPL or EBL). Constraints (6) and (7) identify the used stitch candidates.

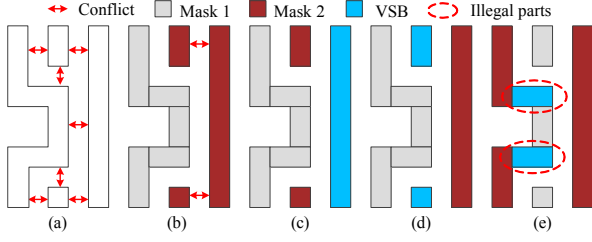


Fig. 4. An example of the post-DPL-decomposition conflict removal. (a) The original layout. (b) The double patterning decomposition result with two unresolvable conflicts. (c) Minimizing the cardinality of VSB to eliminate conflicts. (d) Minimizing the total area of VSB to eliminate conflicts. (e) An illegal solution in which a polygon is generated by two different lithography techniques, i.e., DPL and EBL in this example.

IV. POST-MPL-DECOMPOSITION CONFLICT REMOVAL

In the manufacturing process of the hybrid lithography, the lithography step of MPL is conducted before EBL. Thus, we can regard EBL as an alternative technique for the conflict removal, which is only adopted if necessary. To do that, we can use a two-stage approach, also called the post-MPL-decomposition processing method, which mainly consists of two steps: (1) MPL layout decomposition; (2) Removing a minimum set of nodes to eliminate the unresolved conflicts. One of the advantages of this approach is that the existing state-of-the-art MPL layout decomposition methods can be reused. For example, the existing DPL decomposers [3], [4], [21], TPL decomposers [6]–[9], and MPL decomposer [25] can be used in the first step. However, the negative side of this approach is that the stitch minimization and EBL throughput optimization are conducted in two separate stages, resulting in sub-optimality.

Previous works [15] [16] use the cardinality of VSB as the criterion for EBL utilization. However, the total area of VSB is a more accurate one as discussed in Sec. III-A. We denote the two-stage approach with the criterion of the former as *two-stage-num* and that of the latter as *two-stage-area*.

The two-stage approach for co-optimization of DPL and EBL is illustrated in Fig. 4. Given a layout, we first use the existing DPL decomposer to resolve as many as possible conflicts with the minimum stitch insertion as shown in Fig. 4(b). The results of two-stage-num and two-stage-area are shown in Fig. 4(c) and (d), respectively. Since producing a feature in the layout by two different lithography techniques, i.e., MPL and EBL in our case, will incur greater manufacture cost, we require that a feature should only be produced by either MPL or EBL. For example, the decomposition result shown in Fig. 4(e) is forbidden, although it can eliminate the conflicts.

The first steps of both the two-stage-num and two-stage-area methods are the same. The second steps of the two methods are described in the following.

A. Two-stage-num

For comparison purpose, we develop the two-stage-num method for minimizing the cardinality of VSB. In the second step, we will eliminate the conflicts with a minimum number

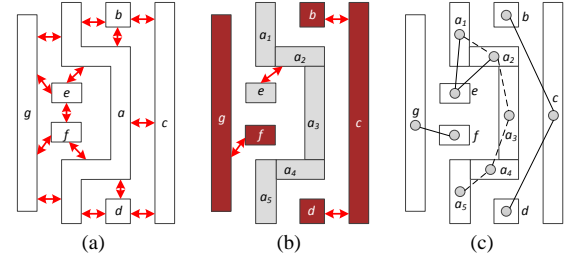


Fig. 5. An example of the remaining conflict graph G_{rem} . (a) The original layout. (b) The double patterning decomposition result. (c) The remaining conflict graph. If the minimum distance of two features assigned to the same mask is less than d_{DP} , then there is a conflict edge (solid line) between the two corresponding nodes in G_{rem} . There is a stitch edge (dashed line) between two features which are from the same polygon.

of VSB. We denote the remaining conflict graph after MPL decomposition as $G_{rem} = (V_r, E_r)$. An example is shown in Fig. 5. Then, we need to find a minimum subset $V'_r \subseteq V_r$, such that for any $(u, v) \in E_r$, either $u \in V'_r$ or $v \in V'_r$. This problem is called a *minimum vertex cover* problem. Note that the minimum vertex cover problem is NP-hard. We use the approximation method in [22], without assigning the area information to each node in the conflict graph.

B. Two-stage-area

We also develop the two-stage-area method for minimizing the total area of VSB. Similar to the method of two-stage-num, in the second step, we need to find a minimum weighted subset of $V'_r \subseteq V_r$, which can cover all the unresolved conflict edges. The weight of each node is calculated according to the area of its corresponding polygon.

V. CONFLICT REMOVAL WITH SIMULTANEOUS STITCH MINIMIZATION AND EBL THROUGHPUT OPTIMIZATION

In this section, we investigate two methods based on a primal-dual framework, one with planarization and one without, to simultaneously improve the EBL throughput and reduce the number of stitches.

A. Co-optimization of DPL Decomposition and EBL Usage

Our first approach mainly consists of three steps: (1) Find a maximum planar subgraph of the original conflict graph. (2) Obtain a set of nodes with minimum total weight to cover all the odd cycles in the planar graph. (3) Add back the non-planar edges and remove a set of nodes with minimum total weight to resolve the remaining conflicts.

1) *Conflict Graph Planarization*: Recall that a graph is planar if and only if it can be drawn in such a way that no edges cross each other. This approach starts with finding a *maximum planar subgraph* of the conflict graph. Note that the maximum planar subgraph problem is NP-hard. Here, we assume that the conflict graph is a nearly planar graph. We first obtain an initial solution using the cactus-tree-based approximation algorithm in [26], which can provide 4/9 performance ratio. To further improve the solution quality, we add the remaining edges greedily using the heuristic in [27], which invokes a

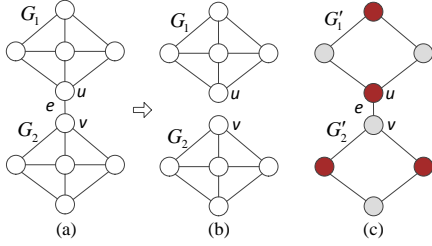


Fig. 6. The bridge removal reduction technique. (a) A connected graph G with a bridge edge e . (b) G can be divided into its 2-edge connected components by removing the bridge e and each subgraph can be solved separately. (c) After solving the VDB problems separately, the merged graph combining the induced bipartite subgraphs is still a bipartite graph.

planarity testing. Besides, previous work [28] proves that the conflict graph is planar with the Manhattan distance. However, DPL problem should be handled by the Euclidean distance due to lithographic nature [10].

2) *Non-cyclable Edge Removal*: Given a planar graph G , we first remove all “non-cyclable” edges, i.e., edges which cannot be part of any cycle in the graph. They include all dangling edges and bridge edges that can be removed without affecting any solution quality. More importantly, this step can ensure that no irregular faces are created in the later process.

Bridge removal has been adopted in TPL decomposition (i.e. the edge deletion K -partition problem for $K = 3$) in previous works [6] [7]. Here, we show that this technique can also be used in the vertex deletion K -partition as well. For the bridge removal reduction technique, we have the following theorem:

Theorem 2. *Solving the MVDB problem with the bridge removal technique will not degrade the quality of the solution. Namely, the number of nodes whose deletion leads to a bipartite graph will not increase with this reduction technique.*

Proof. It is enough to prove that after solving each 2-edge connected component, the solutions of the subcomponents can be combined without degrading the quality.

Without loss of generality, we assume that $G = G_1 \cup G_2 \cup \{e\}$ is a connected graph, where $e = (u, v)$ is the bridge, G_1 and G_2 are the two disconnected components after removing e . Let u be in G_1 and v in G_2 . G'_1 and G'_2 are the induced bipartite graphs corresponding to G_1 and G_2 after removing a set of nodes. Note that e is the only edge between G_1 and G_2 (also between G'_1 and G'_2 if both u and v are retained).

(1) Case one: if both u and v are retained, we only need to prove that the merged graph $G'_1 \cup G'_2 \cup \{e\}$ is still a bipartite graph, i.e., the combination will not incur an odd cycle. Assume that there is an odd cycle C in $G'_1 \cup G'_2 \cup \{e\}$. Since $G'_1 \cup G'_2$ is bipartite, the odd cycle C must contain some vertices in G'_1 and some vertices in G'_2 . Therefore, there are at least two edges between G'_1 and G'_2 due to the odd cycle C , which contradicts that e is the only edge between G'_1 and G'_2 . Thus, $G'_1 \cup G'_2 \cup \{e\}$ is still a bipartite graph. An example is shown in Fig 6.

(2) Case two: If at least one of the end vertices of e is deleted, e will not be in the merged graph. It means that the merged graph $G'_1 \cup G'_2$ is a graph consisting of two disconnected bipartite subgraphs. Therefore, the merged graph is obviously bipartite.

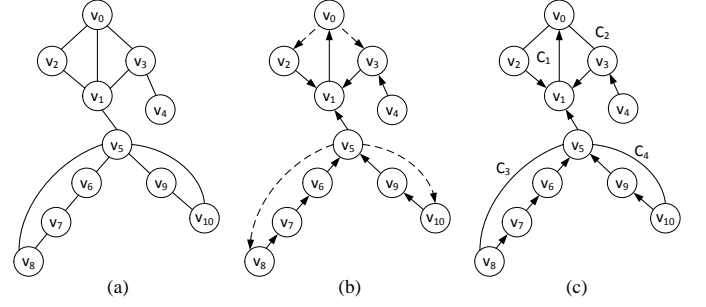


Fig. 7. Chain decomposition. (a) A graph G . (b) The DFS-tree edges (solid lines) and back-edges (dashed lines). (c) A chain decomposition $C = \{C_1, \dots, C_4\}$. The chain C_2 is a path, while the others are cycles. The edge (v_1, v_5) is not contained in any chain, and thus it is a bridge. Similarly, the edge (v_3, v_4) is a dangling edge.

Since solving each 2-edge connected component separately and combining their solutions will not incur a non-bipartite graph, we can conclude that solving MVDB with the bridge removal technique will not degrade the quality of the solution. \square

Note that this technique can be extended to $K > 2$ via color permutations. The proof can be shown in a similar way, which is skipped here. Recently, Schmidt presented a surprisingly simple chain decomposition algorithm for testing on 2-vertex- and 2-edge-connectivity in linear time [18]. Interestingly, we find that the algorithm can also be used for finding all “non-cyclable” edges as well. The algorithm starts with a depth-first-search (DFS), which partition all edges into *tree-edges* (edges in a DFS-tree T) and *back-edges* (edges in G but not in T). Then the graph is traversed again in a specific order starting from the back-edges. At the end of the traversal, all bridges and cut vertices are computed [18]. Since all dangling edges cannot induce any back-edges, they remain *unvisited* during the traversal and hence can also easily be identified and removed. An example is shown in Fig. 7

3) *Minimum Odd Cycle Cover*: In this section, we present some background knowledge of minimum odd cycle cover based on the work of [17]. Consider the following integer linear programming:

$$\begin{aligned} \min \quad & \sum_{v \in V} w_v \cdot x_v, \\ \text{s.t.} \quad & \sum_{v \in C} x_v \geq 1, \quad \forall C \in \mathcal{C} \\ & x_v \in \{0, 1\}, \quad \forall v \in V \end{aligned} \quad (8)$$

where \mathcal{C} is the set of odd cycles, and w_v is the weight of node v . By relaxing the integer constraint $\mathbf{x} \in \{0, 1\}^m$ to $\mathbf{x} \geq 0$, we can get a linear program in the following form:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{w}^T \mathbf{x}, \\ \text{s.t.} \quad & \mathbf{A} \cdot \mathbf{x} \geq \mathbf{p}, \\ & \mathbf{x} \geq 0, \end{aligned} \quad (9)$$

where $n = |\mathcal{C}|$, $m = |V|$, $\mathbf{p}^T = (1, \dots, 1)^n$,

$$a_{ij} = \begin{cases} 1, & \text{the } j\text{-th node is on the } i\text{-th odd cycle,} \\ 0, & \text{otherwise,} \end{cases}$$

a_{ij} is the entry of \mathbf{A} that lies in the i -th row and j -th column. The Lagrangian function with inequality constraints for Problem (9) is

$$L(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}) = \mathbf{w}^T \mathbf{x} + \mathbf{y}^T (\mathbf{p} - \mathbf{A} \mathbf{x}) - \boldsymbol{\lambda}^T \mathbf{x}, \quad (10)$$

where $\mathbf{y} = (y_1, \dots, y_n)^T \geq 0$, $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m)^T \geq 0$ are Lagrangian multipliers. The Lagrangian dual function for the equation (10) is

$$\begin{aligned} g(\mathbf{y}, \boldsymbol{\lambda}) &= \min_{\mathbf{x}} L(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}) \\ &= \min_{\mathbf{x}} (\mathbf{w}^T - \mathbf{y}^T \mathbf{A} - \boldsymbol{\lambda}^T) \mathbf{x} + \mathbf{y}^T \mathbf{p} \\ &= \begin{cases} \mathbf{y}^T \mathbf{p}, & \mathbf{w}^T - \mathbf{y}^T \mathbf{A} - \boldsymbol{\lambda}^T = 0 \\ -\infty, & \text{otherwise.} \end{cases} \end{aligned} \quad (11)$$

Let p^* be the optimum value of Problem (9), and then it is obvious that $p^* \geq L(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}) \geq g(\mathbf{y}, \boldsymbol{\lambda})$. The equation (11) provides a lower bound for p^* . We can obtain the best lower bound from the Lagrangian dual function (11) by solving the following problem,

$$\begin{aligned} \max_{\mathbf{y}, \boldsymbol{\lambda}} \quad & \mathbf{y}^T \mathbf{p}, \\ \text{s.t.} \quad & \mathbf{w}^T - \mathbf{y}^T \mathbf{A} - \boldsymbol{\lambda}^T = 0, \\ & \mathbf{y} \geq 0, \boldsymbol{\lambda} \geq 0. \end{aligned} \quad (12)$$

This problem can also be expressed as,

$$\begin{aligned} \max_{\mathbf{y}} \quad & \mathbf{p}^T \mathbf{y}, \\ \text{s.t.} \quad & \mathbf{A}^T \mathbf{y} \leq \mathbf{w}, \\ & \mathbf{y} \geq 0. \end{aligned} \quad (13)$$

Let d^* be the optimal value of Problem (13). Since the objective and constraints in our problem are linear, we can get the following theorem:

Theorem 3. Problem (9) satisfies the property of the strong duality, i.e., $p^* = d^*$ [29].

Problem (9) is the primal form, and Problem (13) is its dual form. There is a close relationship between the primal and dual, which is called the *complementary slackness* [29]. Namely, if a feasible primal-dual pair x, y is optimal, then we have

- primal slackness: $x_j > 0 \Rightarrow \mathbf{y}^T \mathbf{a}^j = w_j$,
- dual slackness: $y_i > 0 \Rightarrow \mathbf{a}_i \mathbf{x} = p_i$,

where \mathbf{a}^j is the j -th column of \mathbf{A} , and \mathbf{a}_i the i -th row of \mathbf{A} .

4) *Primal-dual Algorithm:* If we relax the primal variable, the dual variable will be reinforced, and vice versa. The complementary slackness can help us to design efficient algorithms based on the idea of approaching the primal problem by improving the dual. In this work, since the graph is planar, we can find a subset of the odd cycles efficiently by computing *faces* in the graph and obtain a minimum vertex cover for the odd-degree faces. After removing the partial vertex cover, we iteratively compute faces again and find another partial odd cycle cover until no odd cycle exists. The face is defined as follows:

Definition 1. A face f of a planar graph consists of a sequence of edges $\{e_0, e_1, \dots, e_{k+1}\}$ and the vertices of the

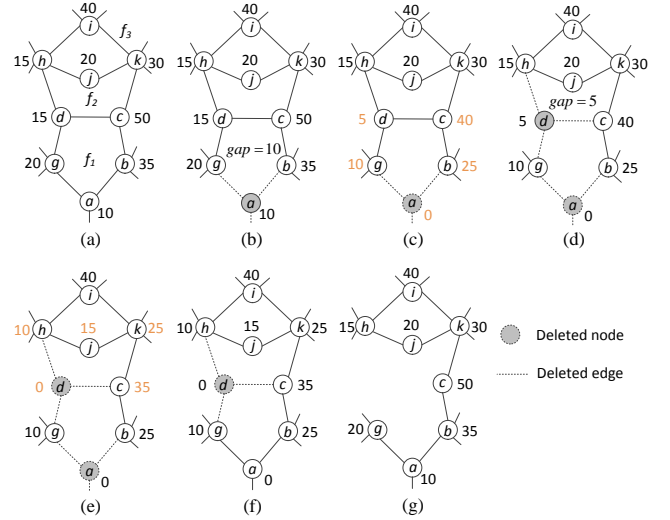


Fig. 8. Illustration of the primal-dual method. (a) The set of odd-degree faces $faceSet = \{f_1, f_2\}$, where $f_1 = (a, b, c, d, g, a)$, $f_2 = (c, k, j, h, d, c)$. $f_3 = (c, k, i, h, d, c)$ is not included in $faceSet$ in the current iteration because we solve smaller odd-degree faces first and f_2 is embedded in f_3 . (b) The first most tight node is a . The gap is 10. (c) The weights of the nodes in f_1 are updated by subtracting 10. (d) The second most tight node is d . The gap is 5. (e) The weights of the nodes in f_2 are updated by subtracting 5. (f) Adding back a will not incur any old cycle, and thus a is retained. (g) Final bipartite graph by removing the minimum weight set of nodes.

edges. Denoting $Inter(f)$ as the interior of f , the following properties are satisfied:

- 1) $source(e_0) = target(e_{k+1})$.
- 2) $face_cycle_pred(e_i) = e_{i+1}$ ($0 \leq i \leq k$).
- 3) $face_cycle_succ(e_{i+1}) = e_i$ ($0 \leq i \leq k$).
- 4) For any different f_1 and f_2 , $Inter(f_1)$ and $Inter(f_2)$ do not intersect except at their boundaries.

where $source(e)$ is the source node of e , $target(e)$ the target node of e , $face_cycle_pred(e)$ the predecessor of e , and $face_cycle_succ(e)$ the successor of e .

An example of computed faces is shown in Fig. 8(a). Next, we compute its odd-degree faces denoted by $faceSet$. Each dual variable y_C ($C \in faceSet$) is initialized as zero. The subroutine $Oracle(S)$ returns the odd-degree faces F not covered by the solution S . Note that, unlike [17] that each node $v \in V$ simply associates a primal variable x_i , here x_i could be shared with a certain number of nodes corresponding to the same polygon. Accordingly, if an x_i is selected as a cover, all vertices that correspond to the same polygon will be selected. To keep track of the nodes from the same polygon, we define a mapping as,

$$VP(u) = \begin{cases} \{u\}, & u \text{ is a virtual node,} \\ \{v \mid v, u \text{ are from the same polygon}\}, & \text{otherwise.} \end{cases}$$

Let $Cyc(VP(v))$ be the set of odd-degree cycles passing through the nodes in $VP(v)$ and let $Q(v) = Cyc(VP(v)) \cap faceSet$ be its subset in the current iteration. We increase y_C for an uncovered odd cycle C_i , where

$$gap(C_i) = \min_{v \in C_i} \{w_v - \sum_{C \in Q(v)} y_C\}, \quad (14)$$

Algorithm 1 Planar_Primal_Dual

Input: $G = (V, E)$: a planar graph
Output: S : a minimum weighted subset of V , covering all the odd cycles

```

1:  $S \leftarrow \emptyset$ 
2:  $\text{gap}(v) \leftarrow w_v, \forall v \in V$ 
3: while an odd cycle exists do
4:   Remove all non-cyclable edges of  $G$ 
5:    $\text{faceSet} \leftarrow \text{compute\_odd\_faces}(G)$ 
6:   while  $\text{Oracle}(S) \neq \emptyset$  do
7:      $F \leftarrow \text{Oracle}(S)$  /* Return odd faces not covered by  $S$  */
8:     for  $C_i \in F$  do
9:        $v_{\text{tight}} \leftarrow \arg \min_{v \in C_i} \{\text{gap}(v)\}$ 
10:      for  $v \in C_i$  do
11:         $\text{gap}(v) \leftarrow \text{gap}(v) - \text{gap}(v_{\text{tight}})$ 
12:      end for
13:       $S \leftarrow S \cup \text{VP}(v_{\text{tight}})$ 
14:    end for
15:  end while
16:  Remove  $S$  and its associated edges from  $G$ 
17: end while
18: /* Post-PD greedy refinement */
19: Sort  $S$  in decreasing order according to the weight
20: for  $v \in S$  do
21:   if  $S \setminus \text{VP}(v)$  is a feasible odd cycle cover then
22:      $S \leftarrow S \setminus \text{VP}(v)$ 
23:   end if
24: end for
```

and the corresponding tight node,

$$v_{\text{tight}} = \arg \min_{v \in C_{\text{tight}}} \{w_v - \sum_{C \in \mathcal{Q}(v)} y_C\}. \quad (15)$$

Note that in the actual implementation, we don't need to explicitly store the dual variables y_C . Instead, a variable named $\text{gap}(v)$ is used for each vertex v to store the cumulative subtraction from the dual variables, where the initialized value is the weight of the vertex:

$$\text{gap}(v) \equiv w_v - \sum_{C \in \mathcal{Q}(v)} y_C. \quad (16)$$

Then $\text{gap}(v)$ is updated by $\text{gap}(v) - \text{gap}(v_{\text{tight}})$ for all $v \in C_i$ and $\text{VP}(v_{\text{tight}})$ is added to S . An example is shown in Fig. 8. Continue executing this until $\text{Oracle}(S)$ is empty. After removing S and the associated edges, we will compute the faces again and re-conduct the above steps iteratively until all odd cycles are covered. To further improve the quality of solution, after sorting S in decreasing order, if $S \setminus \text{VP}(v)$ is a feasible solution, then $\text{VP}(v_{\text{tight}})$ will be removed from S . An example is shown in Fig. 8(e) and (f), where the node a is added back. The details of this algorithm are described in Alg. 1.

5) *Resolving Remaining Conflicts*: After solving the MVD-B problem on the maximum planar subgraph, we add back the non-planar edges, which are removed in the first step. If a non-planar edge does not lead to a conflict, it will be retained. For those conflicting non-planar edges, we construct a remaining conflict graph G_{rem} and find a minimum weighted vertex cover to eliminate the conflicts.

Algorithm 2 Non_Planar_Primal_Dual

Input: $G = (V, E)$: a general graph
Output: S : a minimum weighted subset of V , whose removal will make the remaining graph bipartite

```

1:  $S \leftarrow \emptyset$ 
2:  $\text{gap}(v) \leftarrow w_v, \forall v \in V$ 
3: while  $G$  is not bipartite do
4:   Find an odd cycle  $C$  from  $G$  using a DFS
5:    $v_{\text{tight}} \leftarrow \arg \min_{v \in C} \{\text{gap}(v)\}$ 
6:   for  $v \in C$  do
7:      $\text{gap}(v) \leftarrow \text{gap}(v) - \text{gap}(v_{\text{tight}})$ 
8:   end for
9:    $S \leftarrow S \cup \text{VP}(v_{\text{tight}})$ 
10:  Remove  $\text{VP}(v_{\text{tight}})$  and its associated edges from  $G$ 
11: end while
12: /* Post-PD greedy refinement */
13: Sort  $S$  in decreasing order according to the weight of nodes
14: for  $v \in S$  do
15:   if  $S \setminus \text{VP}(v)$  is a feasible solution then
16:      $S \leftarrow S \setminus \text{VP}(v)$ 
17:   end if
18: end for
```

B. Primal-Dual Method for Non-Planar Graphs

In Sec. V-A, we have introduced a version of primal-dual method, which has also been presented in our preliminary conference paper [1]. It is inspired by the following two aspects: (1) Previous works (e.g. [21], [28]) mentioned the (nearly) planarity of the underlying conflict graphs. (2) It has been proved that the primal-dual algorithm in [17] can obtain 9/4 performance guarantee for planar graphs.

However, the planarization process is quite tricky so that the overall primal-dual method might not be effective. In this section, we introduce a more general primal-dual method as shown in Alg. 2, which is almost the same as the previous method except that it does not require any planarization process. An odd cycle is directly obtained by a depth-first search, which requires a linear run time. However, we found that this non-planar primal-dual method can usually give a better overall runtime performance for the benchmarks in our experiments. The algorithm works as follows: (1) It traverses the graph G to find an odd cycle C (Line 4); (2) Next, it finds the most tight node v_{tight} in C (Line 5); (3) The gaps of each node are updated (Lines 6 to 8); (4) Add $\text{VP}(v_{\text{tight}})$ to the solution and remove it from G (Lines 9 and 10); (5) Continue until G is bipartite. Finally, similar to the planar primal-dual method, a post greedy refinement is conducted to add back the nodes that will not incur any odd cycles.

C. Co-optimization of MPL Decomposition and EBL Usage

In this section, we introduce a random-initialized iterative improvement local search method for the co-optimization of MPL decomposition and EBL throughput. The underlying idea of the local search is rather simple. However, practically it can achieve good results.

Let $P = \{P_1, \dots, P_K, P_{K+1}\}$ be the solution of the co-optimization of MPL and EBL, where P_i ($i = 1, \dots, K$) is the set of nodes assigned to the i -th mask of MPL, P_{K+1} is the node set for EBL. Let $m(P) : P \mapsto \mathbb{R}$ be the measure of

Algorithm 3 Randomized_Iterative_Improvement

Input: $G = (V, E)$: a conflict graph
 K : the number of masks for MPL
Output: P_{best} : the solution of co-optimization of MPL and EBL

```

1: for  $l = 0, \dots, NR_{\text{max}}$  do
2:   Let  $\{P_1^{(l)}, \dots, P_K^{(l)}, P_{K+1}^{(l)}\}$  be a random initialization
3:    $P \leftarrow \{P_1^{(l)}, \dots, P_K^{(l)}, P_{K+1}^{(l)}\}$ 
4:    $\text{Count} \leftarrow 1, i \leftarrow 1$ 
5:   while  $\text{Improve}(P)$  and  $\text{Count} < \text{MaxIter}$  do
6:     Compute  $D(P_i)$ 
7:     if  $m(D(P_i)) < m(P)$  then
8:        $P \leftarrow D(P_i)$ 
9:     end if
10:     $\text{Count} \leftarrow \text{Count} + 1$ 
11:     $i \leftarrow 1 + (\text{Count} \bmod K)$ 
12:  end while
13:  if  $l = 0$  or  $m(P_{\text{best}}) < m(P)$  then
14:     $P_{\text{best}} \leftarrow P$ 
15:  end if
16: end for

```

the solution P . We obtain a neighbor of P by computing the perturbation $D(P_i)$ of P_i , which is defined as follows:

$$D(P_i) = \{P_1, \dots, P'_i, P'_j, \dots, P_K, P'_{K+1}\}, \quad (17)$$

$$j = 1 + (i \bmod K), \quad (18)$$

$$(P'_i, P'_j, P'_{K+1}) = \text{MVDB_Solver}(G_s), \quad (19)$$

$$G_s = (P_i \cup P_j \cup P_{K+1}, \mathcal{E}(P_i \cup P_j \cup P_{K+1})), \quad (20)$$

where $i \in (1, \dots, K)$, $\text{MVDB_Solver}(G_s)$ is the MVDB solver, $\mathcal{E}(P_i \cup P_j \cup P_{K+1})$ is the set of edges whose vertices are in $P_i \cup P_j \cup P_{K+1}$.

We first start from a random initialization P and search among the neighbors of P to find the *local optimum*. To jump out of the local optimum, we use several random initializations and search among their respective neighbors. Finally, we choose the best solution from the search space. The details are described in Alg. 3. NR_{max} is the maximum number of initializations and MaxIter is the maximum number of iterations after each initialization. An example result of this method is shown in Fig. 9. We choose a connected component from the conflict graph of the case C432 [6] with $d_{MP} = 160\text{nm}$ and $K = 3$. We use 15 random initializations. Area is the total area of the features that are produced by e-beam. The convergence process of the first random initialization is shown in Fig. 9(a), where given the initialization the solution converges to its local optimum fast after several iterations. The respective local optimums of the 15 random initializations are shown in Fig. 9(b). At last, we choose the best local optimum as the final solution.

VI. EXPERIMENTAL RESULTS

Our proposed method for the co-optimization of the hybrid lithography combining MPL and EBL is implemented in C++. All of the experiments are performed on a Linux workstation with 3GHz CPU and 4GB memory. The ISCAS-85 & 89 benchmarks from the authors of [6] are utilized to test the performance of our methods. The minimum wire length and the minimum wire spacing is 30nm and 50nm, respectively.

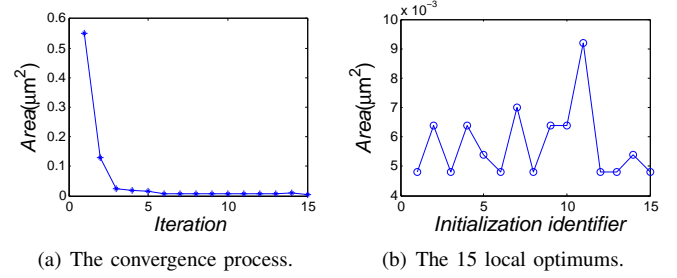


Fig. 9. An example result of the random-initialized local search method. Area is the total area of the features produced by EBL. (a) The iteration process after the first initialization. (b) The respective local optimums after the 15 random initializations.

TABLE III
STATISTICS OF THE CASES WITH $d_{DP} = 120\text{nm}$.

Cases	#Polygon	#Conf	#Node	#Edge	#NPE	ratio
C432	1033	2937	2883	4787	273	5.70%
C499	2134	6494	5536	9896	681	6.88%
C1355	2963	8390	8915	14342	610	4.25%
C3540	9910	25026	26274	41390	1899	4.59%
C5315	14235	37184	38523	61472	3541	5.76%
C7552	20490	52846	55677	88034	4153	4.72%
S38417	66182	142127	144501	220465	6942	3.15%
S35932	150137	354564	342529	546956	20957	3.83%
S38584	162792	346718	355001	538932	17626	3.27%
S15850	155508	347250	349210	540952	19340	3.58%

Instead of scaling down the layouts, we increase the minimum distance d_{MP} to simulate the situation of the sub-22nm circuits. α and β are set to 1 and 0.01, respectively.

A. Results of DPL Decomposition and EBL Throughput

1) *Comparison with the Two-stage Methods:* Since none of the existing work focuses on the co-optimization of stitch minimization and EBL throughput, we implemented the two-stage methods in Sec. IV as the baseline. To show that minimizing the total area of VSB is more accurate, two-stage-num (Sec. IV-A) and two-stage-area (Sec. IV-B) were implemented, respectively. The existing DPL solver in [21] was adopted in the first stage.

The statistics of the cases with $d_{DP}=120\text{nm}$ are in Table III, where #Polygon is the number of polygons, #Conf the initial conflict number, #Node and #Edge the number of nodes and edges in the conflict graph, and #NPE the number of non-planar edges computed by the method in Sec. V-A1. ratio is the percentage of non-planar edges, defined as $\text{ratio} = \frac{\#NPE}{\#Edge}$. The histogram of ratio for different d_{DP} is given in Fig. 10.

The results are in Table IV, where ‘‘Planar PD’’ refers to the simultaneous co-optimization method in Sec. V-A. A and #VSB correspond to the total area and total number of VSB. #S denotes the number of double patterning stitches. Comparing with the two-stage-num, the two-stage-area method reduced the total area for EBL by 13% on average, although its #VSB is larger than that of two-stage-num on average. Since the throughput of EBL is determined by A , minimizing the total area of VSB is thus more accurate in the real situation.

Comparing with the two-stage-area, the primal-dual method can dramatically reduce the total area of VSB by 65.5% on

TABLE IV
CO-OPTIMIZATION RESULTS OF DPL STITCH MINIMIZATION AND EBL UTILIZATION WITH $d_{DP} = 120nm$.

Cases	Two-stage-num				Two-stage-area				Planar PD			
	$A(\mu m^2)$	#VSB	#S	Time(s)	$A(\mu m^2)$	#VSB	#S	Time(s)	$A(\mu m^2)$	#VSB	#S	Time(s)
C432	1.87	328	9	0.14	1.35	339	17	0.13	0.58	251	4	0.14
C499	4.62	867	18	0.57	4.24	875	15	0.57	1.91	753	5	0.51
C1355	6.16	787	59	0.28	5.25	783	63	0.28	1.69	630	17	0.34
C3540	29.73	2781	174	1.14	26.60	2856	178	1.14	7.99	2677	79	1.32
C5315	37.25	3655	366	1.62	32.10	3667	421	1.61	11.38	3571	126	1.88
C7552	53.74	5680	340	2.07	45.41	5603	361	2.07	14.91	4948	140	2.41
S38417	183.31	18624	1184	4.39	156.38	18844	1195	4.40	53.14	16743	384	4.55
S35932	425.22	46742	3074	13.39	379.21	47252	3172	13.43	133.32	39583	954	13.54
S38584	427.00	45525	3554	11.57	364.36	46266	3543	11.58	132.60	40654	1005	11.61
S15850	465.13	46723	3252	11.75	410.40	46785	3290	11.72	134.65	41864	1060	11.98
Avg.	3.32	1.13	3.19	0.97	2.90	1.14	3.25	1.00	1.00	1.00	1.00	1.00

TABLE V
COMPARISON BETWEEN ILP METHOD AND PRIMAL DUAL METHOD WITH $d_{DP} = 100nm$.

Cases	ILP				Planar PD				Non-planar PD			
	$A(\mu m^2)$	#VSB	#S	Time(s)	$A(\mu m^2)$	#VSB	#S	Time(s)	$A(\mu m^2)$	#VSB	#S	Time(s)
C432	0.51	227	8	1.56	0.51	229	8	0.09	0.54	237	9	0.01
C499	1.55	684	49	34957.80	1.60	688	43	0.37	1.56	682	40	0.01
C1355	1.15	486	68	4.83	1.19	506	67	0.19	1.21	503	64	0.02
C3540	5.22	2184	502	266.44	5.57	2235	468	0.97	5.63	2255	434	0.06
C5315	7.31	3072	775	111.71	7.81	3132	727	1.20	7.75	3106	657	0.09
C7552	10.21	4175	888	5778.89	10.84	4267	854	1.59	10.75	4234	812	0.13
S38417	NA	NA	NA	>7714Min	36.56	13265	2089	3.68	35.87	13203	2034	0.39
S35932	NA	NA	NA	>7714Min	101.22	33033	4899	11.28	98.20	32499	5031	1.03
S38584	NA	NA	NA	>7714Min	95.93	33181	5377	9.64	94.80	32965	5451	1.03
S15850	NA	NA	NA	>7714Min	98.94	35179	6040	9.70	97.44	35062	5948	1.05
Avg.	NA	NA	NA	NA	1.02	1.01	1.00	10.13	1.00	1.00	1.00	1.00

Note: NA - Not available, Min - Minutes.

TABLE VI
PLANAR PD WITH DIFFERENT MAXIMUM PLANAR SUBGRAPH FINDING ALGORITHMS ($d_{DP} = 100nm$).

Cases	CT + Planar PD				GCT + Planar PD			
	$A(\mu m^2)$	#VSB	#S	Time(s)	$A(\mu m^2)$	#VSB	#S	Time(s)
C432	0.65	246	12	0.03	0.51	229	8	0.09
C499	2.58	822	88	0.06	1.60	688	43	0.37
C1355	1.35	547	83	0.08	1.19	506	67	0.19
C3540	7.23	2505	528	0.25	5.57	2235	468	0.97
C5315	11.26	3636	861	0.38	7.81	3132	727	1.20
C7552	14.25	4714	1013	0.54	10.84	4267	854	1.59
S38417	58.77	15703	2631	1.53	36.56	13265	2089	3.68
S35932	164.19	40175	6826	3.83	101.22	33033	4899	11.28
S38584	141.99	38518	6874	3.72	95.93	33181	5377	9.64
S15850	160.75	41919	7679	3.80	98.94	35179	6040	9.70
Avg.	1.56	1.18	1.29	0.37	1.00	1.00	1.00	1.00

average with more or less the same runtime. Besides, the primal-dual method can also significantly reduce the number of stitches by 69.2% on average. Note that less total area for VSB means a higher throughput for EBL and that less stitches mean lower manufacturing cost for DPL. Thus, our primal-dual method can simultaneously improve the throughput and reduce the manufacturing cost comparing with the two-stage methods. Therefore, the experimental results further demonstrate the effectiveness of our proposed primal-dual method.

Note that DPL+EBL is different from TPL. In TPL, there are *unresolvable conflicts* as shown in Fig. 11(c), which can make the whole chip failed. In DPL+EBL, there are no unresolvable conflicts (assuming the writing resolution of EBL is sufficient) because all the unresolvable conflicts can be resolved by e-beam direct writing.

Fig. 12 shows the DPL+EBL decomposition result for C432.

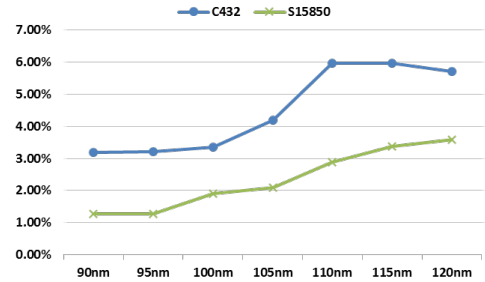


Fig. 10. Histogram of the percent of non-planar edges for different d_{DP} .

2) *Comparison with the ILP-based Method:* In this experiment, we compare our primal-dual method with the ILP-based

TABLE VII
THE INFLUENCE OF STITCH INSERTION WITH $d_{DP} = 100nm$.

Cases	$\beta = 0.01$				$\beta = 5$				$\beta = 100$			
	$A(\mu m^2)$	#VSB	#S	Time(s)	$A(\mu m^2)$	#VSB	#S	time(s)	$A(\mu m^2)$	#VSB	#S	Time(s)
C432	0.54	237	9	0.01	0.56	244	1	0.01	0.57	247	0	0.01
C499	1.56	682	40	0.01	1.69	694	3	0.01	1.72	702	0	0.02
C1355	1.21	503	64	0.02	1.42	555	7	0.02	1.50	582	0	0.01
C3540	5.63	2255	434	0.06	7.10	2469	62	0.06	7.63	2655	0	0.06
C5315	7.75	3106	657	0.09	9.52	3275	123	0.09	10.56	3641	0	0.10
C7552	10.75	4234	812	0.13	13.47	4710	72	0.13	14.06	4911	0	0.13
S38417	35.87	13203	2034	0.39	42.01	14431	177	0.37	43.52	14930	0	0.39
S35932	98.20	32499	5031	1.03	114.48	35808	142	0.99	115.75	36175	0	1.02
S38584	94.80	32965	5451	1.03	112.06	36424	228	1.02	113.88	37005	0	1.02
S15850	97.44	35062	5948	1.05	116.27	38453	314	1.02	119.02	39370	0	1.06
Avg.	1.00	1.00	1.00	1.00	1.18	1.10	0.06	0.97	1.21	1.12	0.00	1.00

TABLE VIII
CO-OPTIMIZATION RESULTS OF TPL STITCH MINIMIZATION AND EBL UTILIZATION WITH $d_{TP} = 160nm$.

Cases	Two-stage-num				Two-stage-area				RILS + Planar PD				
	$A(\mu m^2)$	#VSB	#S	Time(s)	$A(\mu m^2)$	#VSB	#S	Time(s)	$A(\mu m^2)$	#VSB	#S	Time(s)	#NR
C432	0.46	108	17	5.00	0.34	107	18	5.14	0.15	82	17	8.83	5
C499	1.40	333	46	17.19	1.02	335	46	17.61	0.53	288	43	20.96	5
C1355	0.99	207	96	10.74	0.77	212	96	10.45	0.34	157	94	30.35	5
C3540	2.78	574	380	45.21	1.61	583	381	45.34	0.92	541	320	44.89	5
C5315	5.21	1013	415	69.83	2.85	1050	415	69.98	1.78	1012	378	65.04	5
C7552	8.10	1321	640	94.06	4.26	1346	646	93.20	2.23	1219	646	100.58	5
S38417	20.66	4634	2403	315.44	13.15	4701	2422	322.06	8.06	4714	2278	252.02	4
S35932	60.24	13963	7184	1117.94	39.86	14111	7237	1115.12	25.21	14118	6788	1067.84	4
S38584	46.21	10957	6132	773.70	29.53	11117	6167	773.20	19.23	11392	5840	605.89	4
S15850	67.80	14292	6983	800.00	41.21	14390	7048	810.74	24.27	14568	6318	694.98	4
Avg.	2.59	0.99	1.07	1.12	1.63	1.00	1.08	1.13	1.00	1.00	1.00	1.00	-

TABLE IX
COMPARISON BETWEEN RILS + PLANAR PD AND RILS + NON-PLANAR PD WITH $d_{TP} = 160nm$.

Cases	RILS + Planar PD					RILS + Non-planar PD				
	$A(\mu m^2)$	#VSB	#S	Time(s)	#NR	$A(\mu m^2)$	#VSB	#S	Time(s)	#NR
C432	0.15	82	17	8.83	5	0.16	85	11	0.41	5
C499	0.53	288	43	20.96	5	0.55	298	33	0.79	5
C1355	0.34	157	94	30.35	5	0.37	174	60	1.22	5
C3540	0.92	541	320	44.89	5	0.97	555	286	3.16	5
C5315	1.78	1012	378	65.04	5	1.89	981	307	4.54	5
C7552	2.23	1219	646	100.58	5	2.34	1220	548	6.83	5
S38417	8.06	4714	2278	252.02	4	8.52	4924	1883	15.24	4
S35932	25.21	14118	6788	1067.84	4	27.26	15039	5273	46.64	4
S38584	19.23	11392	5840	605.89	4	20.20	11859	5029	37.41	4
S15850	24.27	14568	6318	694.98	4	26.12	15423	5139	41.82	4
Avg.	0.94	0.95	1.22	18.29	-	1.00	1.00	1.00	1.00	-

method (Sec. III-D). We use GLPK [30] as the ILP solver. The bridge removal reduction technique (Sec. V-A2) is used to reduce problem size. The comparison results are shown in Table V. “Planar PD” is the method in Sec. V-A and “Non-planar PD” is the one in Sec. V-B.

From Table V, we can see that the ILP-based method is very sensitive to the structure of conflict graphs. If the conflict graphs can be reduced into smaller components by the bridge removal technique, the problem size can also be reduced and thus the runtime can be very small, like the cases C432 and C1355. However, if the graphs cannot be reduced, the runtime can be very large, like the cases C499, C3540, and C7552. Besides, for the last four larger cases, we even cannot obtain any results within several days.

Comparing with the ILP-based method, our planar PD method can obtain reasonably good results for the first six

cases (C432 to C7552), while the runtime can be significantly reduced on average. For the last four larger cases, the planar PD method can still deliver results efficiently.

Comparing with the planar PD method, our non-planar PD method can achieve similar or even slightly better results, while the runtime can be improved significantly. The time saving is mainly due to the fact that no planarization is performed in the non-planar PD method.

3) *Planar PD with Different Planarization Algorithms*: We compare two maximum planar subgraph finding algorithms for our planar primal-dual method (Sec. V-A). The results are shown in Table VI. “CT” denotes the cactus-tree-based algorithm in [26]. “GCT” denotes the algorithm that uses the solution of CT as initialization and then adds the remaining edges greedily using the heuristic in [27]. As shown in Table VI, CT + planar PD can run about 3X faster than GCT +

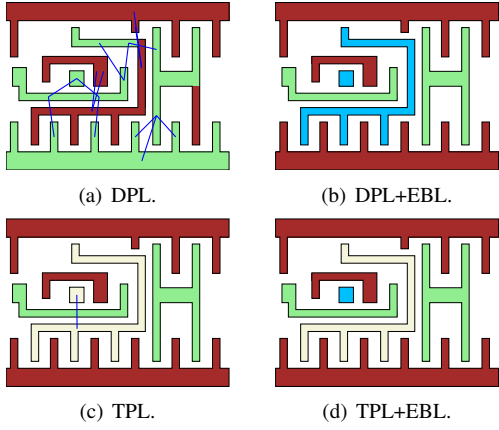


Fig. 11. Comparison between DPL, DPL+EBL, TPL, and TPL+EBL for a 45nm cell. The blue color is for EBL, while the others are for different masks of MPL. The blue line indicates unresolved conflicts. $d_{MP}=200\text{nm}$.

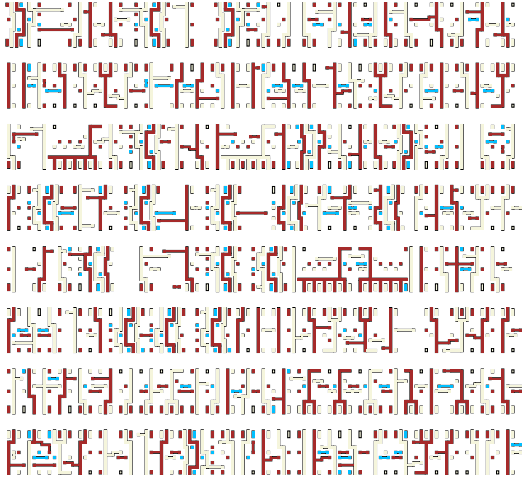


Fig. 12. The DPL+EBL decomposition result for C432. The blue color is for EBL. The other two colors are for different masks of DPL.

planar PD on average, while the solution quality of GCT + planar PD can be much better than that of CT + planar PD. A better planarization algorithm usually can directly benefit our planar PD method. Therefore, we expect that if there is a planarization algorithm as fast as CT while its solution quality can match or outperform GCT, our planar PD method may achieve better runtime performance.

4) *Influence of Stitch Insertion*: In this experiment, we study the influence of stitch insertion in the DPL+EBL decomposition. The e-beam area weight α is fixed to 1. The stitch weight β is set to 0.01, 5, and 100, respectively. The results are shown in Table VII. We use the non-planar PD method (Sec. V-B) as the MVDB solver. $d_{DP}=100\text{nm}$.

We can see that if β is set to 0.01 (i.e., the stitch weight is very small compared with the e-beam area weight), stitches will be sacrificed to save e-beam usage. However, when β is set to 100 (i.e., the stitch weight is very large compared with the e-beam area weight), no stitches will be used and the total area of e-beam will be increased by 21% on average to solve the conflicts in the layout. When β is set to a value between 0.01 and 100, the primal-dual method will make a tradeoff to

solve the conflicts according to the weights of each other. Note that different foundries or hybrid lithography systems might have different requirement on the stitch weight. In practice, the weights should be set by the designers or manufacturers to control the manufacturing cost and improve the throughput simultaneously.

B. Results of TPL Decomposition and EBL Throughput

1) *Comparison with the Two-stage Methods*: With the decreased feature size, DPL is not sufficient. Thus TPL is proposed as shown in Fig. 11(c). We implemented the version of the two-stage methods for the hybrid lithography of TPL and EBL. The existing TPL decomposer in [9] is used in the first stage. The detailed results are reported in Table VIII. “RILS” denotes the random-initialized local search method in Sec. V-C, and #NR denotes the number of initializations. Comparing with the two-stage-num, the two-stage-area method reduced the total area for EBL by more than 37% on average, although its #VSB is larger than that of two-stage-area except the case C432. Thus, this result further verifies that minimizing A is more accurate. Comparing with the two-stage-area, our proposed RILS with the planar PD method can further reduce the total area for EBL by 38.7% on average and can reduce the number of stitches by 7.4% on average simultaneously. The runtime of our proposed RILS method is more or less the same as the two stage method. Therefore, the experimental results demonstrate the effectiveness of the proposed RILS method.

Note that our proposed framework can handle with constrained decomposition for the hybrid lithography of MPL and EBL by pre-processing the conflict graph and color permutations. The constrained decomposition requires that certain features in the layout should be assigned to a certain mask, which is very useful in a practical flow. The co-optimization result of a 45nm layout with PG straps by our RILS with the planar PD method is shown in Fig. 13, where the P&G net is constrained to be assigned to Mask 1.

2) *RILS with Different MVDB Solvers*: In this experiment, we compare the RILS + planar PD method with the RILS + non-planar PD method. The results are shown in Table IX. #NR is the number of random initializations. From the table, we can see that the RILS + non-planar PD method can achieve more or less the same results comparing with the RILS + planar PD method. The total e-beam area is increased by 6% on average, while the number of stitches is decreased by 18% on average. However, the runtime of the RILS + non-planar PD method is improved significantly.

VII. CONCLUSION

The hybrid lithography combining LELE-style MPL and EBL is promising with the decreased feature size. We introduce a new layout decomposition framework for the MPL+EBL hybrid lithography, which considers the stitch minimization and EBL throughput simultaneously. To do that, we have proposed the planar primal-dual method for the co-optimization of DPL and EBL, and the random-initialized local search method for that of MPL and EBL. Experimental results demonstrate the efficiency of our proposed methods.

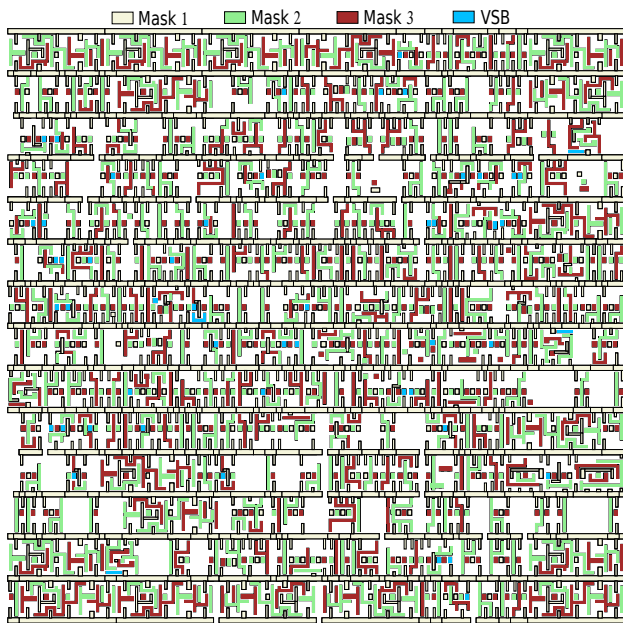


Fig. 13. The co-optimization of TPL and EBL for a 45nm layout. The P&G net is constrained to be assigned to Mask 1.

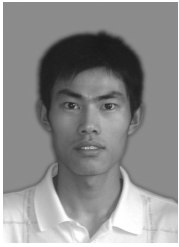
We have also introduced a non-planar primal-dual method which can achieve more or less the same quality of results while the runtime can be improved significantly. We have also discussed how to reduce the stage movement and settling time to further improve the throughput on a typical EBL system.

ACKNOWLEDGEMENT

The authors would like to thank Prof. Yifang Chen from State Key Lab. of ASIC and System in Fudan University for providing the EBL system data and calculating the e-beam exposure time. We would like to thank Prof. Walter Hu from UT Dallas for helpful discussion about EBL system. We would also like to thank the anonymous reviewers for their constructive comments.

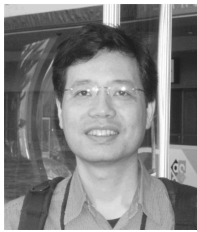
REFERENCES

- [1] Y. Yang, W.-S. Luk, H. Zhou, C. Yan, X. Zeng, and D. Zhou, "Layout decomposition co-optimization for hybrid e-beam and multiple patterning lithography," in *IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)*, 2015.
- [2] K. Yuan, B. Yu, and D. Z. Pan, "E-beam lithography stencil planning and optimization with overlapped characters," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 31, no. 2, pp. 167–179, 2012.
- [3] A. Kahng, C.-H. Park, X. Xu, and H. Yao, "Layout decomposition approaches for double patterning lithography," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 6, pp. 939–952, 2010.
- [4] K. Yuan, J.-S. Yang, and D. Z. Pan, "Double patterning layout decomposition for simultaneous conflict and stitch minimization," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 2, pp. 185–196, 2010.
- [5] X. Tang and M. Cho, "Optimal layout decomposition for double patterning technology," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2011, pp. 9–13.
- [6] B. Yu, K. Yuan, D. Ding, and D. Pan, "Layout decomposition for triple patterning lithography," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 3, pp. 433–446, 2015.
- [7] S.-Y. Fang, Y.-W. Chang, and W.-Y. Chen, "A novel layout decomposition algorithm for triple patterning lithography," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 33, no. 3, pp. 397–408, 2014.
- [8] J. Kuang and E. F. Young, "An efficient layout decomposition approach for triple patterning lithography," in *ACM/IEEE Design Automation Conference (DAC)*, 2013.
- [9] Y. Zhang, W.-S. Luk, H. Zhou, C. Yan, and X. Zeng, "Layout decomposition with pairwise coloring for multiple patterning lithography," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2013.
- [10] S.-Y. Fang, S.-Y. Chen, and Y.-W. Chang, "Native-conflict and stitch-aware wire perturbation for double patterning technology," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 31, no. 5, pp. 703–716, 2012.
- [11] J. Sun, Y. Lu, H. Zhou, and X. Zeng, "Post-routing layer assignment for double patterning," in *IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)*, 2011, pp. 793–798.
- [12] S. Steen, S. McNab, L. Sekaric, I. Babich, J. Patel, J. Buchignano, M. Rooks, D. Fried, A. Topol, J. Brancaccio *et al.*, "Looking into the crystal ball: future device learning using hybrid e-beam and optical lithography (keynote paper)," in *Micro lithography 2005*. International Society for Optics and Photonics, 2005, pp. 26–34.
- [13] 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 *IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)*, 2012, pp. 707–712.
- [14] Y. Ding, C. Chu, and W.-K. Mak, "Throughput optimization for SADP and e-beam based manufacturing of 1D layout," in *ACM/IEEE Design Automation Conference (DAC)*, 2014, pp. 1–6.
- [15] J.-R. Gao, B. Yu, and D. Z. Pan, "Self-aligned double patterning layout decomposition with complementary e-beam lithography," in *IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)*, 2014, pp. 143–148.
- [16] H. Tian, H. Zhang, Z. Xiao, and M. D. Wong, "Hybrid lithography for triple patterning decomposition and e-beam lithography," in *SPIE Advanced Lithography*. International Society for Optics and Photonics, 2014, pp. 90 520P–90 520P.
- [17] M. X. Goemans and D. P. Williamson, "Primal-dual approximation algorithms for feedback problems in planar graphs," *Combinatorica*, vol. 18, no. 1, pp. 37–59, 1998.
- [18] J. M. Schmidt, "A simple test on 2-vertex- and 2-edge-connectivity," *Information Processing Letters*, vol. 113, no. 7, pp. 241–244, 2013.
- [19] S. Rizvi, *Handbook of photomask manufacturing technology*. CRC Press, 2005.
- [20] N. W. Parker, A. D. Brodie, and J. H. McCoy, "High-throughput NGL electron-beam direct-write lithography system," in *Micro lithography 2000*. International Society for Optics and Photonics, 2000, pp. 713–720.
- [21] W.-S. Luk and H. Huang, "Fast and lossless graph division method for layout decomposition using SPQR-tree," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2010, pp. 112–115.
- [22] G. Ausiello and *et al*, *Complexity and Approximability Properties: Combinatorial Optimization Problems and Their Approximability Properties*. Springer, 1999.
- [23] F. Hadlock, "Finding a maximum cut of a planar graph in polynomial time," *SIAM Journal on Computing*, vol. 4, no. 3, pp. 221–225, 1975.
- [24] H.-A. Choi, K. Nakajima, and C. S. Rim, "Graph bipartization and via minimization," *SIAM Journal on Discrete Mathematics*, vol. 2, no. 1, pp. 38–47, 1989.
- [25] B. Yu and D. Z. Pan, "Layout decomposition for quadruple patterning lithography and beyond," in *ACM/IEEE Design Automation Conference (DAC)*, 2014, pp. 1–6.
- [26] G. Călinescu, C. G. Fernandes, U. Finkler, and H. Karloff, "A better approximation algorithm for finding planar subgraphs," *Journal of Algorithms*, vol. 27, no. 2, pp. 269–302, 1998.
- [27] T. Poranen, "Two new approximation algorithms for the maximum planar subgraph problem," *Acta Cybern.*, vol. 18, no. 3, pp. 503–527, 2008.
- [28] Y. Xu and C. Chu, "A matching based decomposer for double patterning lithography," in *Proceedings of the 19th international symposium on Physical design*. ACM, 2010, pp. 121–126.
- [29] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*. Courier Dover Publications, 1982.
- [30] A. Makhorin, "GLPK - the GNU linear programming toolkit," <http://www.gnu.org/directory/GNU/glpk.html>, 2014.



Yunfeng Yang received the B.S. degree in information engineering from Southeast University, Nanjing, China, in 2008. He is currently pursuing the Ph.D. degree with State Key Lab. of Application Specific Integrated Circuits and System, Microelectronics Department, Fudan University, Shanghai, China.

He was with Synopsys, Shanghai, during the summer of 2013, 2014, and 2015 as a software engineering R&D intern. His current research interests include design for manufacturability, mathematical modeling, and optimization.



Wai-Shing Luk was graduated from Chinese University of Hong Kong in 1988 with a B.Sc. degree in Electronics. He received his M.Phil. and Ph.D. degrees in Computer Science and Engineering from the same university in 1993 and 1996 respectively. In 1997, he was awarded a postdoctoral fellowship at Katholieke Universiteit Leuven in Belgium. He worked as a Senior R&D Engineer at Synopsys in US from 1999. After joining Fudan University as an Associate Professor in 2004, he has published over 30 papers in international conferences and reviewed

journals. His research interests include design for manufacturability, statistical analysis and optimization, and VLSI physical design algorithms.



David Z. Pan (S'97-M'00-SM'06-F'14) received the B.S. degree from Peking University, Beijing, China, and the M.S. and Ph.D. degrees from the University of California, Los Angeles, CA, USA. From 2000 to 2003, he was a Research Staff Member at IBM T. J. Watson Research Center, Yorktown Heights, NY, USA. He is currently the Engineering Foundation Professor with the Department of Electrical and Computer Engineering, The University of Texas (UT) at Austin, Austin, TX, USA. His current research interests include nanoscale design

for manufacturability and reliability, physical design, vertical integration design and technology, and design/CAD for emerging technologies. He has published over 230 papers in refereed journals and conferences, and holds eight U.S. Patents.

He has served as a Senior Associate Editor for ACM Transactions on Design Automation of Electronic Systems (TODAES), an Associate Editor for IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems (TCAD), IEEE Transactions on Very Large Scale Integration Systems (TVLSI), IEEE Transactions on Circuits and Systems PART I (TCAS-I), IEEE Transactions on Circuits and Systems PART II (TCAS-II), Science China Information Sciences (SCIS), Journal of Computer Science and Technology (JCST), IEEE CAS Society Newsletter, etc. He has served in the Executive/Program Committees of many major conferences, including Design Automation Conference (DAC), International Conference on Computer Aided Design (ICCAD), Asia and South Pacific Design Automation Conference (ASPDAC), and International Symposium on Physical Design (ISPD).

He was the recipient of the SRC 2013 Technical Excellence Award, the DAC Top Ten Author in Fifth Decade Award, the DAC Prolific Author Award, the Asia and South Pacific Design Automation Conference Frequently Cited Author Award in 2015, 12 Best Paper Awards at premier venues such as ASPDAC, Design, Automation & Test in Europe (DATE), ICCAD, IBM Research, ISPD, and SRC TECHCON, various international CAD Contest awards, the Communications of the ACM Research Highlights in 2014, the ACM/SIGDA Outstanding New Faculty Award in 2005, the NSF CAREER Award in 2007, the SRC Inventor Recognition Award thrice, the IBM Faculty Award four times, the UCLA Engineering Distinguished Young Alumnus Award in 2009, the UT Austin RAISE Faculty Excellence Award in 2014, the UCLA Computer Science Department Outstanding PhD Award in 2000, the ACM Recognition of Service Award in 2007 and 2008, among others. From 2008 to 2009, he was an IEEE CAS Society Distinguished Lecturer.



Hai Zhou (SM'04) received the B.S. and M.S. degrees in computer science and technology from Tsinghua University, Beijing, China, in 1992 and 1994, respectively, and the Ph.D. degree in computer sciences from the University of Texas, Austin, in 1999. He is currently an Associate Professor of Electrical Engineering and Computer Science with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL. His current research interests include very large scale integration computer-aided design, algorithm design,

and formal methods. Dr. Zhou was a recipient of the CAREER Award from the National Science Foundation in 2003.



Changhao Yan received the B.E. and M.E. degrees in fluid mechanics and computer science from the Huazhong University of Science and Technology, Wuhan, China, in 1996 and 2002, respectively, and the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2006. Currently, he is a Lecturer with State Key Lab. of Application Specific Integrated Circuits and Systems, Microelectronics Department, Fudan University, Shanghai, China. His current research interests include parasitic parameter extraction of interconnects, design for manufacturability, chemical mechanical polishing modeling, and simulation.



Dian Zhou (M'89-SM'07) received the B.S. degree in physics and M.S. degree in electrical engineering from Fudan University, China, in 1982 and 1985, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Illinois in 1990.

He joined the University of North Carolina at Charlotte as an assistant professor in 1990, where he became an associate professor in 1995. He joined the University of Texas at Dallas as a full professor in 1999. His research interests include: High-speed

VLSI systems, CAD tools, mixed-signal ICs, and algorithms.

Dr. Zhou received the Research Initiation Award from National Science Foundation in 1991, the IEEE Circuits and Systems and Society Darlington Award in 1993, and the National Science Foundation Young Investigator Award in 1994. He also served as a panel member of the NSF CAREER Award in 1996. He was a Guest Editor for the International Journal of Custom-Chip Design, Simulation and Testing, and was an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS from 1996 to 1998. He received Chinese NSF Overseas Outstanding Young Scientist Award in 2000, and Chinese Yangzi River Scholar from 2002 to 2007. He was the panel member of "Moving up the Technology Chain" at World Economy Forum, Davos, Switzerland, 2006. He received Changjiang Honor Professor from Fudan University in 2008, and was selected as "Thousand People Plan" professor from China in 2011.



Xuan Zeng (M'97) received the B.Sc. and Ph.D. degrees in electrical engineering from Fudan University, Shanghai, China, in 1991 and 1997.

She is currently a Full Professor with the Microelectronics Department and serves as the Director of State Key Laboratory of ASIC and Systems, Fudan University, Shanghai, China. She was a Visiting Professor with the Electrical Engineering Department, Texas A&M University, College Station, and Microelectronics Department, Technische Universiteit Delft, Delft, The Netherlands, in 2002

and 2003. Her research interests include design for manufacturability, high-speed interconnect analysis and optimization, analog behavioral modeling, circuit simulation, and ASIC design.

Dr. Zeng was a recipient of the first-class Award of Electronic Information Science and Technology from the Chinese Institute of Electronics in 2005. She received the second-class Award of Science and Technology Advancement and the Cross-Century Outstanding Scholar Award from the Ministry of Education of China in 2006 and 2002. She received the award of IT Top 10 in Shanghai in 2003. She served on the Technical Program Committee of the IEEE/Association for Computing Machinery Asia and South Pacific Design Automation Conference in 2000 and 2005.