Yield-aware Decomposition for LELE Double Patterning

Yukihide Kohira[a], Yoko Yokoyama[b], Chikaaki Kodama[b],
Atsushi Takahashi[c], Shigeki Nojima[b] and Satoshi Tanaka[b]

[a]The University of Aizu, Aizu-Wakamatsu, Japan
[b]Toshiba Corporation, Yokohama, Japan
[c]Tokyo Institute of Technology, Tokyo, Japan

ABSTRACT

In this paper, we propose a fast layout decomposition algorithm in litho-etch-litho-etch (LELE) type double patterning considering the yield. Our proposed algorithm extracts stitch candidates properly from complex layouts including various patterns, line widths and pitches. The planarity of the conflict graph and independence of stitch-candidates are utilized to obtain a layout decomposition with minimum cost efficiently for higher yield. The validity of our proposed algorithm is confirmed by using benchmark layout patterns used in literatures as well as layout patterns generated to fit the target manufacturing technologies as much as possible. In our experiments, our proposed algorithm is 7.7 times faster than an existing method on average.

Keywords: Double Patterning, LELE, Desing for Manufacturability

1. INTRODUCTION

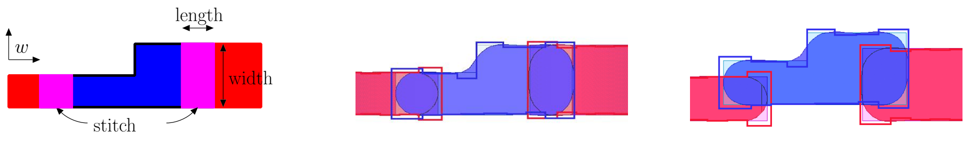
LELE type double patterning[1] which seems to be most practical solution for the 22 nm node enables us to fabricate smaller features without using advanced technologies such as extreme ultraviolet (EUV) lithography. Recently, although triple patterning is often discussed, it is not easy to adopt it due to issues related to manufacturability such as misalignment of masks, and the extra mask cost of triple patterning does not seem affordable when double patterning can achieve the same pitch. In LELE type double patterning, a layout pattern is decomposed and assigned to two masks so that each can be formed on a wafer by an exposure. The yield which affects manufacturing cost much depends on a layout pattern decomposition. A layout pattern decomposition method needs to have an ability to obtain a layout pattern decomposition which achieves higher yield.

In layout pattern decomposition for LELE type double patterning, a component in a layout pattern can be partitioned into smaller features and features can be assigned to distinct masks. Adjoining features assigned to different masks are requested to be overlapped to take an overlay error into account. The overlap area between adjoining features which are assigned to different masks is called a *stitch*. For example, a component is partitioned into three features and assigned to red and blue masks as shown in Fig. 1(a). In the figure, a magenta rectangle represents a stitch. The existence of a stitch affects the yield since the actual wafer images are degraded even if OPC (optical proximity correction) is applied. For example, wafer images obtained by lithography simulation after OPC without and with overlay error are shown in Fig. 1(b) and (c), respectively. A layout decomposition without using stitches is preferred. However, stitches are often essential in decompositions. Moreover, no decomposition often exists even if stitches are used. A decomposition is characterized by a set of stitches used in the decomposition, called *stitch selection*. Our problem is to find an optimum stitch selection that achieves higher yield.

In order to find a better layout pattern decomposition, various methods have been proposed so far.[2–16] A pattern segmentation method to identify stitch-candidates is proposed by Yang et al.[7] The segmentation method identifies stitch-candidates effectively but misses some stitch-candidates when a variety of widths of line patterns is large. A variety of widths depends on target layer and a variety of widths of some layers is large.

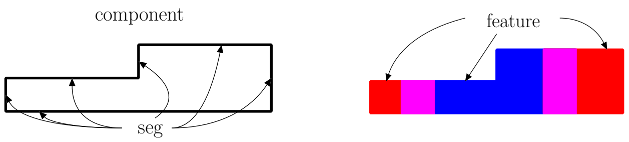
Further author information: (Send correspondence to Yukihide Kohira, Yoko Yokoyama and Atsushi Takahashi)
Yukihide Kohira: E-mail:kohira@u-aizu.ac.jp, Telephone: +81-242-37-2536
Yoko Yokoyama: E-mail:yoko.yokoyama@mail.semicon.toshiba.co.jp, Telephone: +81-45-890-2818
Atsushi Takahashi: E-mail:atsushi@eda.ce.titech.ac.jp, Telephone: +81-3-5734-2665

Design-Process-Technology Co-optimization for Manufacturability VIII, edited by John L. Sturtevant,
Luigi Capodieci, Proc. of SPIE Vol. 9053, 90530T · © 2014 SPIE
CCC code: 0277-786X/14/$18 · doi: 10.1117/12.2046180



(a) target wafer image (b) lithography simulation with OPC (c) lithography simulation with OPC and overlay

Figure 1. Stitch and wafer image.



(a) before decomposition (b) after decomposition

Figure 2. Am example of a component and features.

An efficient polynomial time algorithm to obtain a layout pattern decomposition by using the minimum number of stitches in terms of given stitch-candidates is proposed by Tang and Cho.[13] However, the yield is not taken into consideration but the minimum number of stitches is. In order to find a minimum cost stitch selection, a matching based method is proposed by Xu and Chu.[9] The method utilizes the planarity of a constraint graph, and detects faces of the constraint graph which are needed to be broken by stitches to obtain a feasible layout pattern decomposition. These faces are matched by using a minimum cost stitch selection and are broken. However, how a stitch is inserted is not well-defined in the method especially when a variety of widths of line patterns is large.

The minimum stitch length is usually given in the design rule. Although the higher yield is achieved if the minimum stitch length is set to large enough, there is a trade-off between the yield and the existence of decomposition. A stitch satisfying the minimum length constraint may cause a hotspot. As deduced from examples shown in Fig. 1(b) and (c), a narrow short stitch is more fragile than a wide long stitch, and the yield degradation caused by a stitch is mitigated by increasing the length or area of the stitch.

In this paper, we propose an fast algorithm which obtains a pattern decomposition with minimum cost to achieve higher yield for LELE double patterning. First, stitch-candidates which help not to degrade the quality of wafer image are extracted. In our proposed algorithm, one stitch-candidate is defined in each interval where a stitch can be inserted without violating the design rule. A stitch-candidate may cross each other when the width of a line pattern is large, but is independent of other stitch-candidates, that is, no design rule violation occurs in any stitch selection obtained from the defined stitch-candidates. Then, a minimum cost stitch selection which derives a feasible yield-aware layout pattern decomposition is obtained from the stitch-candidates. By using the cost of a stitch which reflects the quality degradation on wafer image appropriately, a layout pattern decomposition which maximizes the yield is obtained by a minimum cost stitch selection. In our method, a stitch-candidate is defined when its length can be larger than or equal to the minimum length, and the cost of a stitch is defined by using the area of a stitch after assumed overlay error to reflect the yield degradation.

2. PRELIMINARIES

In the layout area, components in one layer are given as input. A component is a maximal connected region in the layout area (Fig. 2(a)). In layout pattern decomposition, a component is partitioned into smaller features with overlap, and a feature is assigned to one of two masks (Fig. 2(b)). The overlap area between adjoining features which are assigned to different masks is called a *stitch*. A line segment between two adjacent corners of a boundary of a feature or component is called a *seg*. A seg is either horizontal or vertical and distinct seg neither touch nor cross each other. A stitch is rectangular region which disconnects a component. In order to specify the position of a stitch, the centerline of the stitch is used. The centerline of a stitch is a horizontal or vertical slice-line of a component. The orthogonal stitches may overlap each other.

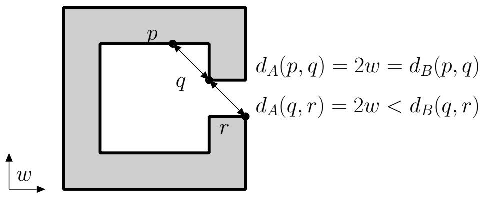


Figure 3. Dp-spacing is satisfied between p and q but not between q and r ($w_d = 3w$).

In this paper, Manhattan distance is used to define our assumed design rule. In case that Euclid distance is used in actual specification, a threshold value in Manhattan distance is set approximately to meet the specification, or Euclid distance is used by ignoring false violations around corners of components. Let $d_A(p,q)$ and $d_B(p,q)$ be the distances between points p and q in the layout area and in components, respectively. Note that $d_B(p,q)$ is the minimum length of routes connecting p and q in components. If there is no route between them, the distance is defined infinite. Apparently, $d_A(p,q) \leq d_B(p,q)$. Let $d_A(P,Q)$ and $d_B(P,Q)$ be the distances between objects P and Q in the layout area and in components, respectively. The minimum length of the horizontal or vertical slice-line of a component (feature) is called a *width* of a component (feature).

The design rule used is described by using four constraints, called *width*, *spacing*, *double-patterning-spacing* (*dp-spacing*), and *stitch-length*. The width constraint is violated if and only if there is a feature or component whose width is less than w_w. The spacing constraint is violated if and only if there is a pair of points p and q in components such that $d_A(p,q) < w_s$ and $d_A(p,q) < d_B(p,q)$. The dp-spacing constraint is violated if and only if there is a pair of points p and q in components in a mask such that $d_A(p,q) < w_d$ and $d_A(p,q) < d_B(p,q)$. The stitch-length constraint is violated if and only if the length of a stitch is less than w_t. In Fig. 3, when $w_d = 3w$, points p and q satisfy the dp-spacing constraint, but points q and r do not. Although different threshold values can be used in our design rule as used in practice, a simplest design rule is used to explain the concept of our method.

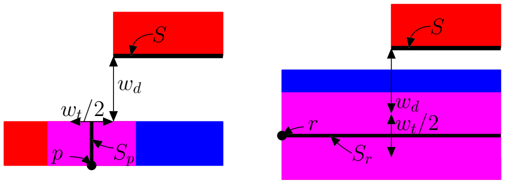
In the following, we focus on a layout pattern which satisfies width and spacing constraints. A layout pattern decomposition, a decomposition hereinafter, is said to be *legal* if width, spacing, dp-spacing, and stitch-length constraints are satisfied.

Stitch-candidates are defined by partitioning a component into features. A stitch-candidate becomes a stitch in a decomposition if both sides of it are assigned to different masks in the decomposition. A set of stitches which defines a decomposition is called a *stitch selection*. The stitch selection without stitches is called the *empty stitch selection*. A stitch selection is said to be *legal* if a legal decomposition is obtained by it. A positive *cost* is set to each stitch-candidate which reflects the impact on the quality of wafer image. The cost of a stitch selection is the sum of costs of stitches in the selection. Our problem is to find a minimum cost legal stitch selection.

3. PROPOSED ALGORITHM

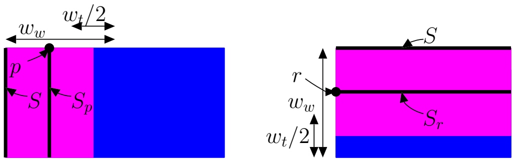
The outline of our proposed decomposition algorithm which finds a minimum cost legal stitch selection is as follows:

1. Extract all the stitch-candidates in the layout pattern. Construct the *conflict graph*.
2. For each connected component in the conflict graph, the following procedures are applied.
   (a) Construct the *relation graph* of the empty stitch selection, and specify faces for the focused connected component.
   (b) If the focused connected component is not bipartite, then report odd faces of the relation graph in order to help to modify the layout pattern.
   (c) If the number of odd faces is zero, then obtain a decomposition without stitches.



(a) Too close to the adjacent component (b) Invalid stitch-candidate on major axis of thick line

Figure 4. Points p and r are on the alley wall.



(a) Orthogonal to the major axis (b) Parallel to the major axis

Figure 5. Points p and r are on the corner wall.

(d) Construct the *face graph* from the relation graph of the empty stitch selection, and then construct the *cost graph* from the face graph by finding shortest paths between odd faces.

(e) Specify a minimum cost legal stitch selection by finding a minimum-cost perfect matching in the cost graph, and obtain a decomposition that corresponds to the specified minimum cost legal stitch selection.

3. Merge decompositions for all connected components in the conflict graph, and output them.

In our proposed method, a speedup technique[13] in which a conflict graph is decomposed into connected components is adopted. Hereinafter, each step is discussed in detail by using a connected component in the conflict graph.

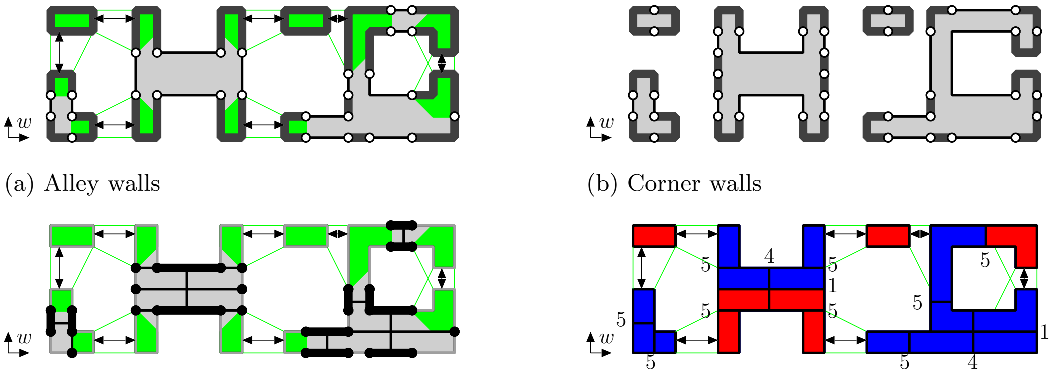
4. STITCH-CANDIDATE

In this section, stitch-candidates which are generated in our proposed algorithm are defined. Our proposed algorithm excludes stitch-candidates that violate the design rule. First, two types of locations where stitch is not allowed are discussed. Then, the cost of stitch used in this paper is defined.

First type called an *alley wall* is defined on the boundary of a component to prevent from generating stitch-candidates at narrow spacing location. If the distance between boundaries of components is less than w_d, then no stitch which cuts these boundaries can be used without violating the dp-spacing constraint. Formally, alley walls are defined as the set of points p on boundaries of components such that the slice-line S_p of a component which ends at p has a seg S such that $d_A(S_p, S) < w_d + w_t/2$ and $d_A(S_p, S) < d_B(S_p, S)$. For example, points p and r shown in Fig. 4 are contained in alley wall.

Second type called a *corner wall* is defined on the boundary of a component to prevent from generating stitch-candidates which generate a short feature. If a stitch is near an end of line pattern, then a short feature which violates the minimum width constraint is generated. In order to satisfy the minimum width constraint and the minimum stitch length constraint, the centerline of a stitch cannot be located near an end of line pattern. Formally, corner walls are defined as the set of points p on boundaries of components such that the slice-line S_p of a component which ends at p has a seg S of the same component which is parallel to S_p and $d_A(S_p, S) < w_w - w_t/2$. For example, points p and r shown in Fig. 5 are contained in corner wall.

Corner wall and alley wall are simply called *wall.* Obviously, when the centerline of a stitch whose length is minimum cuts a wall, the stitch is not contained in any legal stitch selection. On the other hand, a stitch which does not cut a wall can be used in a legal stitch selection without violating design rule when its length is



(a) Alley walls

(b) Corner walls

(c) Intervals and stitch-candidates

(d) Features, stitch cost, and decomposition of cost 6

Figure 6. Features, stitch-candidates, and decomposition with two stitches ($w_w = w_s = w$, $w_d = 3w$, $w_t = 0$).

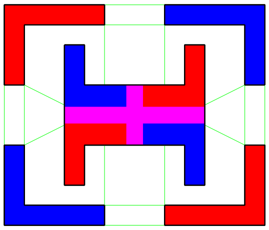


Figure 7. An example of layout that requires orthogonal crossing stitches in feasible decomposition.

minimum. In our algorithm, an interval which consists of maximal parallel boundaries defined by excluding walls is used to define a stitch-candidate. A slice-line which connects the pair of parallel outer boundaries of an interval is used as the centerline of a stitch. By defining one stitch-candidate for each interval, stitches are independent of other stitches when these lengths are minimum. Although the position of the centerline of a stitch is arbitrary within each interval, the centerline is set to the center of the interval so that the length of the stitch can be increased as much as possible if necessary without affecting other stitches. A *feature* is defined as a maximum connected region of a component obtained by cutting the component by the centerlines of stitch-candidates.

For example, the alley walls and corner walls are shown in Fig. 6(a) and (b), respectively. They are represented by thick gray lines. In Fig. 6(a), a green region represents area where a stitch cannot pass without violating the dp-constraint. In Fig. 6(c), the outer boundaries of intervals where a stitch can be inserted without violating the design rule are represented by thick black lines. An end of each interval is closed. A line connecting outer boundaries of an interval represents the centerline of a stitch-candidate. The numbers of features and stitch-candidates are 20 and 13, respectively. In Fig. 6(d), the cost of each stitch-candidate just for reference and an example of decomposition into red and blue using two stitches are shown. A stitch-candidate is a straight line segment inside a component connecting the boundary of the component. The number of stitches in this decomposition is minimum and the cost is 6.

Note that our proposed method extracts the stitch-candidate properly even if orthogonal stitches overlap each other. In an example shown in Fig. 7, a feasible decomposition requires orthogonal crossing stitches. Our proposed method finds a feasible decomposition shown in Fig. 7, but existing methods cannot since they cannot handle orthogonal crossing stitches.

The cost of stitch used in this paper is defined as follows. Let $l(s)$ and $w(s)$ be the length and width of a stitch (or a stitch-candidate) s, respectively. $l(s)$ is defined as the sum of w_t and the length of the interval at which s is defined. In order to take the maximum possible stitch length in evaluation, the effective length $l'(s)$ and width $w'(s)$ of s are defined as $l'(s) = \min\{l(s), L_{\max}\}$ and $w'(s) = \min\{w(s), W_{\max}\}$, respectively, where $L_{\max}$ and $W_{\max}$ are user-defined thresholds. Let $c(s)$ be the cost of a stitch (or a stitch-candidate) s. In this

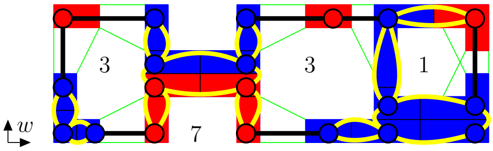


Figure 8. Relation graph and decomposition with two stitches.

paper, the cost is defined as

$$c(s) = (L_{\max} W_{\max} - (l'(s) - M)(w'(s) - M)) + C,$$

where M is the maximum expected misalignment of the mask caused by overlay error and C is a big constant. The effective area of s given as $(l'(s) - M)(w'(s) - M)$ is defined as the area of s when the misalignment is maximum. The effective area of s is used to reflect the yield degradation. The cost is smaller if the length (width) is larger when the length (width) is at most $L_{\max}$ ($W_{\max}$). If C is large enough, then the minimum number of stitches is guaranteed while the sum of effective areas is maximized.

5. LEGAL STITCH SELECTION

In this section, the legality of stitch selection is discussed by defining conflict graph and relation graph. In order to obtain a legal stitch selection, the *conflict graph* which represents the relation between features is defined. A vertex in the conflict graph corresponds to a feature, and an edge is added between vertices which correspond to features with narrow spacing which cannot be assigned to the same mask simultaneously. Formally, an edge is added between vertices corresponding to features b_i and b_j if and only if there is a pair of points $p \in b_i$ and $q \in b_j$ such that $d_A(p, q) < w_d + w_t/2$ and $d_A(p, q) < d_B(p, q)$. A layout pattern has a legal stitch selection if and only if the conflict graph of the layout pattern is bipartite.

A decomposition is not unique in general. Our problem is to obtain a minimum cost stitch selection which corresponds to a decomposition. If the empty stitch selection is legal, then it is the minimum cost legal stitch selection. Although it is easy to check whether the empty stitch selection is legal, it is not trivial for an arbitrary stitch selection. In order to check whether a stitch selection is legal, the *relation graph* is defined from the conflict graph by adding edges. In the relation graph, an edge that corresponds to a boundary of a component which is cut by a stitch-candidate not contained in the stitch selection is added between corresponding vertices. In the relation graph, edges corresponding to edges in the conflict graph are called *conflict-edges*, and the others are called *boundary-edges*. A cycle in a relation graph is said to be *illegal* if the number of conflict-edges in the cycle is odd. A stitch selection of a layout pattern is legal if and only if the relation graph of the stitch selection contains no illegal cycle, that is, the number of conflict-edges in every cycle is even.

A minimum cost legal stitch selection is a stitch selection such that the number of conflict-edges in every cycle in the relation graph is even and that the total cost is minimum. In Fig. 8, the relation graph of the empty-stitch selection of the layout pattern shown in Fig. 6 is shown. Black and yellow edges represent conflict-edges and boundary-edges, respectively. This relation graph contains illegal cycles, but they are broken if 4 boundary-edges which corresponds to two stitches in the shown decomposition are removed.

6. ODD FACE OF LAYOUT PATTERN

The planarity of relation graph helps to find a minimum cost legal stitch selection, though the conflict graphs defined for double patterning are not planar in general.[13] However, when $w_s < w_d \leq w_w + 2w_s$ and $w_t \leq w_w$, there is a natural embedding of a relation graph onto plane where the embedding of each edge corresponds to a shortest path that does not pass the other components. We focus on a layout pattern where edges do not cross in such embedding, and a plane embedding of a relation graph is used in the following.

Let G_r be a plane embedding of a relation graph. A face of G_r that does not have conflict-edge corresponds to features of the layout pattern. In the following, a face of G_r with at least one conflict-edge is focused. The

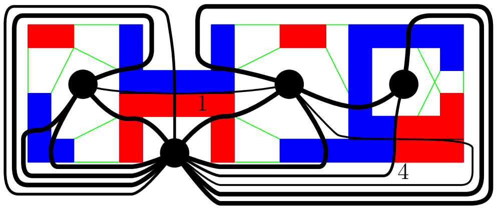


Figure 9. Face graph and the minimum cost legal stitch selection with decomposition of cost 5.

parity of a face of G_r is defined as the parity of the number of conflict-edges in the edge set, counting bridges twice. The number of conflict-edges in each face is shown in Fig. 8. The number of odd faces of G_r is even. Note that if G_r contains no odd face, then the number of conflict-edges in every cycle of G_r is even and the corresponding stitch selection is legal. When a stitch is added to the stitch selection, the boundary-edges which correspond to the stitch are removed from G_r, and two faces of G_r are merged into one. A minimum cost legal stitch selection is obtained from the empty stitch selection by adding stitches with minimum cost so that all the odd faces of G_r are eliminated.

In order to characterize a minimum cost legal stitch selection, the *face graph* which represents the relation between faces is defined. A node in the face graph corresponds to a face of G_r with at least one conflict-edge where G_r is a plane embedding of the relation graph of the empty stitch selection. An edge in the face graph is inserted between vertices corresponding to faces connected by a stitch-candidate. The weight of an edge is positive and corresponds to the cost of the corresponding stitch-candidate. The multiple edges between two nodes may be merged if preferred. A minimum cost legal stitch selection corresponds to a set of paths connecting odd faces which eliminates all the odd faces whose total cost is minimum. In order to find a minimum cost legal stitch selection, the *cost graph* of a layout pattern is defined. A node in the cost graph corresponds to an odd face of the relation graph of the empty stitch selection. An edge in the cost graph is inserted between vertices if and only if there is a path connecting them in the face graph. The cost of an edge is the weight of a shortest path connecting them in the face graph. If the conflict graph is bipartite, then a perfect matching exists in the cost graph. A stitch is contained at most once in a minimum-cost perfect matching in the cost graph. A stitch selection is a minimum cost legal stitch selection if and only if it corresponds to a minimum-cost perfect matching of the cost graph.

For example, the face graph of the layout pattern shown in Fig. 6 is shown in Fig. 9. The number of nodes in the face graph is 4. All faces that have conflict-edges are odd. The weight of an edge corresponds to the cost of the corresponding stitch. The cost of a minimum-cost perfect matching in the cost graph is 5. The decomposition corresponding to the minimum cost stitch selection is also shown.

7. EXPERIMENTS

7.1 Comparison with Existing Methods

We implemented the proposed method in C++ language, and the method is executed on a Linux machine with 6 GB memory by using single Intel core i7-940 of 2.93 GHz.

First, our method is evaluated by comparing with state-of-the-art methods in terms of the number of stitches. ISCAS benchmarks used in Ref. 7 and 13 are used which are reproduced from the information given by authors of Ref. 7 and 13 and from figures in Ref. 7, though we could not obtain the same data. The benchmark data are shown in Table 1. In Table 1, #comp, #seg, #ce, #cc, #fc, #ofc, #sc, and #st represent the number of components (patterns), the number of segs, the number of conflict-edges, the number of connected components in the conflict graph, the number of faces, the number of odd faces, the number of stitch-candidates, and the number of stitches used that is equal to the minimum number of stitches in a legal decomposition, respectively.

As mentioned in section 3, a speedup technique is adopted. To show the efficiency of a speedup technique, we apply both the proposed method with it (Ours w. speedup) and without it (Ours w/o. speedup). In this experiment, the cost $c(s)$ of stitch s is set to 1, and we followed the parameters as in in Ref. 7 and 13 which are w_d = 54 nm and w_t = 20 nm. The number of stitches by our method is same as the other methods for

Table 1. ISCAS benchmarks.

name	#comp	#seg	#ce	#cc	#fc	#ofc	#sc	#st
c432	850	4918	540	414	518	2	1028	1
c499	1491	9518	1489	502	1002	100	2067	50
c880	1872	10666	1422	717	984	344	2472	198
c1355	2656	15246	1514	1328	1514	164	3290	114
c1908	4191	24370	3141	1733	2416	418	5267	371
c2670	6371	37564	5802	2056	3543	1350	8769	947
c3540	8188	47244	6897	2896	4501	1622	10914	1034
c5315	11498	68476	10097	3926	6451	2464	16464	1545
c6288	11605	64762	5602	6259	6515	512	15354	256
c7552	17167	99526	14027	6258	9376	3046	22590	2058

#comp	the number of components (patterns)
#seg	the number of segs
#ce	the number of conflict-edges
#cc	the number of connected components in the conflict graph
#fc	the number of faces
#ofc	the number of odd faces
#sc	the number of stitch-candidates
#st	the number of stitches

Table 2. Comparison on computation time. (ILP,[7]Tang:[13] 3.0 GHz, 4 GB memory, Ours: 2.93 GHz, 6 GB memory)

	ILP[7]	Tang[13]		Ours w/o. speedup		Ours w. speedup	
name	tot(s)	tot(s)	sol(s)	tot(s)	sol(s)	tot(s)	sol(s)
c432	0.63	0.23	0.03	0.07	<0.01	0.02	<0.01
c499	100.0	0.47	0.03	0.10	<0.01	0.05	<0.01
c880	4525.6	0.44	0.03	0.12	0.01	0.09	<0.01
c1355	702.4	0.49	0.02	0.17	<0.01	0.08	<0.01
c1908	37019.8	1.10	0.04	0.39	0.01	0.16	<0.01
c2670	>24Hr	2.00	0.11	0.83	0.13	0.37	0.01
c3540	>24Hr	2.64	0.14	1.20	0.21	0.44	0.02
c5315	>24Hr	4.55	0.24	2.27	0.48	0.69	0.02
c6288	>24Hr	3.23	0.25	1.40	0.03	0.29	<0.01
c7552	>24Hr	8.19	0.32	4.14	0.79	0.89	0.03
ave.		7.71		2.91		1	

each layout pattern. In Table 2, results are summarized. "tot" is the total execution time and "sol" is the time excluding the time to construct the relation graph and to output the result. "ave." is the average ratio of the total execution time of the method in Ref. 13 and our method without the speedup technique by our method with the speedup technique. The results of ILP and Tang in Table 2 are directly copied from Ref. 13. The our proposed method with a speedup technique is 7.7 times faster than the method in Ref. 13 in which several speedup techniques are adopted on average. Note also that our method without speedup technique is 2.8 times faster than the method in Ref. 13 on average. We believe that the impact on computation time caused by the difference of experimental setting is not big since a machine used in our experiments seems to be inferior than machines used in literatures and was available at the time when literatures were published.

Table 3. Layout pattern by Open Cell Library.

name	#comp	#seg	#ce	#sc	#st	tot(s)	sol(s)
55x55	12626	93580	13907	36231	776	0.948	0.014

#comp	the number of components (patterns)
#seg	the number of segs
#ce	the number of conflict-edges
#sc	the number of stitch-candidates
#st	the number of stitches

Table 4. Evaluation of stitch selections.

condition	#st	len (μm)	width (μm)	area (μm^2)	area-e (μm^2)
Min-stitch	776	15.520	62.755	1.256	1.178
Min-stitch len-exp	776	54.785	62.755	4.359	4.242
Max-length	776	56.184	61.340	4.402	4.286
Max-area (Ours)	776	56.169	68.315	4.955	4.831

Table 5. The cumulative distribution of area of stitches ([%]).

method	overlay	1	2	3	4	5	6	7	8	9	10	11
Min-stitch	0%	92.5	98.2	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Min-stitch len-exp		1.9	8.5	24.2	82.0	87.8	89.2	93.7	96.8	97.6	97.7	97.8
Max-length		1.0	6.3	21.8	83.8	89.7	90.2	94.1	97.3	97.7	97.7	97.7
Max-area (Ours)		1.0	6.3	21.8	74.0	80.7	82.3	87.6	93.3	93.7	93.9	93.9
Min-stitch	15%	93.7	98.7	99.2	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Min-stitch len-exp		3.5	13.3	49.1	58.5	87.0	90.9	94.1	95.4	97.2	97.6	98.1
Max-length		2.4	14.9	51.9	61.3	87.6	91.1	94.2	96.0	97.7	97.8	98.2
Max-area (Ours)		2.2	13.9	48.5	56.3	78.2	84.8	88.9	91.0	93.9	94.2	94.7

7.2 Minimization of Costs

Next, the quality of decomposition is evaluated by using the cost of stitch selection. A decomposition obtained by our method with the speedup technique is compared with other decompositions. A layout pattern is generated based on PDKv1_3_v2010_12 in Nangate FreePDK45 Library.[17] Among several layers in the library, Metal1 is used for evaluation. Since the library is generated without considering the double-patterning, patterns are modified to fit the double-patterning technology as much as possible. In order to validate the cost, it is necessary to calculate overlap area around stitch. However, simple shrink causes too many hotspots and we find the layout improvement are required for this evaluation. We can improve the design rule of this layout and eliminate hotspots but, in this case, it is difficult to judge whether the contribution comes from DR improvement or our method. We decide to use the layout at 40 nm generation for this validation and apply appropriate decomposition rules and lithography conditions. The parameters are set as follows: $w_w = w_s = 65$ nm, $w_d = 70$ nm, $w_t = 20$ nm, $L_{\max} = 80$ nm, $W_{\max} = 400$ nm, $M = 1$ nm, $C = 10L_{\max} \cdot W_{\max}$ nm^2. In this setting, the minimum number of stitches is not guaranteed, but the number of stitches is minimum in experiments.

In Tables 3, 4, and 5, results are summarized. In these tables, "Min-stitch" is a decomposition with the minimum number of stitches. Since the existing methods[9,13] only take the minimum number of stitches into consideration, the results by the existing methods may be similar to results of "Min-stitch". "Min-stitch len-exp" is obtained from a decomposition with the minimum number of stitches by maximizing the length of each stitch as much as possible. "Max-length" is obtained by using $(L_{\max} - (l'(s) - M)) + C$ as the cost function. "Max-area" is the result of our method. The statistics of the layout pattern and the result of our method are shown in Table 3. In Table 4, "len", "width", "area", and "area-e" represent the sum of stitch lengths $l'(s)$, the sum of stitch widths $w'(s)$, the sum of stitch areas $l'(s)w'(s)$, and the sum of stitch areas after maximum misalignment $(l'(s) - M)(w'(s) - M)$, respectively. In Table 5, the cumulative distribution of area of stitches is summarized. Stitches are categorized by areas which are obtained by lithography simulation. Note that the area of a stitch may increase depending on the direction of misalignment. For each condition, the worst case among eight directions is used. Under the existence of overlay error, the number of stitches which have small area increase much. It is confirmed that a decomposition of our method is robuster than other decomposition since the number of stitches with small area is smallest. Although the overlay error used is a practical value, the numerical data is not fit for double patterning technology. In order to get more reasonable comparisons, layout patterns that are compliant to double patterning technology are required.

In practical, the maximization of the total area may not contribute to increase the yield but the maximization of the minimum area of a stitch may do. Our proposed method can be enhanced easily to the maximization of the minimum area of a stitch by introducing the threshold area. For example, edges with less than the threshold area are ignored in the face graph, and the threshold area is maximized by checking the existence of a feasible layout pattern decomposition by our proposed method.

8. CONCLUSIONS

In this paper, we propose an efficient algorithm which gives a feasible layout decomposition for litho-etch-litho-etch (LELE) type double patterning with the minimum cost. The correctness and efficiency of the proposed algorithm are theoretically guaranteed, and the validity of our implementation is confirmed by experiments. In our experiments, our proposed algorithm is 7.7 times faster than an existing method on average. The quality of decomposition in terms of the lithograph compliance depends on the cost assigned to each stitch-candidate. The definition of cost of stitches which maximizes the total yield by taking mask density balance, stitch direction, and etc. into account is in our future works.

ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant-in-Aid for Scientific Research (B) 25280013.

REFERENCES

[1] Bailey, G. E., Tritchkov, A., Park, J.-W., Hong, L., Wiaux, V., Hendrickx, E., Verhaegen, S., Xie, P., and Versluijs, J., "Double pattern EDA solutions for 32nm HP and beyond," in [*Proc. SPIE*], **6521**, 65211K (2007).

[2] Kahng, A., Park, C.-H., Xu, X., and Yao, H., "Layout decomposition for double patterning lithography," in [*Proc. ICCAD*], 465–472 (2008).

[3] Cho, M., Ban, Y., and Pan, D. Z., "Double patterning technology friendly detailed routing," in [*Proc. ICCAD*], 506–511 (2008).

[4] Yuan, K., Yang, J.-S., and Pan, D., "Double patterning layout decomposition for simultaneous conflict and stitch minimization," in [*Proc. ISPD*], 107–114 (2009).

[5] Hsu, C.-H., Chang, Y.-W., and Nassif, S. R., "Simultaneous layout migration and decomposition for double patterning technology," in [*Proc. ICCAD*], 595–600 (2009).

[6] Xu, Y. and Chu, C., "GREMA: Graph reduction based efficient mask assignment for double patterning technology," in [*Proc. ICCAD*], 601–606 (2009).

[7] Yang, J.-S., Lu, K., Cho, M., Yuan, K., and Pan, D. Z., "A new graph-theoretic, multi-objective layout decomposition framework for double patterning lithography," in [*Proc. ASP-DAC*], 637–644 (2010).

[8] Yuan, K., Yang, J.-S., and Pan, D., "Double patterning layout decomposition for simultaneous conflict and stitch minimization," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **29**, 185–196 (2 2010).

[9] Xu, Y. and Chu, C., "A matching based decomposer for double patterning lithography," in [*Proc. ISPD*], 121–126 (2010).

[10] Yuan, K. and Pan, D. Z., "WISDOM: Wire spreading enhanced decomposition of masks in double patterning lithography," in [*Proc. ICCAD*], 32–38 (2010).

[11] Chen, S.-Y. and Chang, Y.-W., "Native-conflict-aware wire perturbation for double patterning technology," in [*Proc. ICCAD*], 556–561 (2010).

[12] Hsu, C.-H., Chang, Y.-W., and Nassif, S. R., "Simultaneous layout migration and decomposition for double patterning technology," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **30**, 284–294 (2 2011).

[13] Tang, X. and Cho, M., "Optimal layout decomposition for double patterning technology," in [*Proc. ICCAD*], 9–13 (2011).

[14] Ghaida, R. S., Agarwal, K. B., Nassif, S. R., Yuan, X., Liebmann, L. W., and Gupta, P., "A framework for double patterning-enabled design," in [*Proc. ICCAD*], 14–20 (2011).

[15] Fang, S.-Y., Chen, S.-Y., and Chang, Y.-W., "Native-conflict and stitch-aware wire perturbation for double patterning technology," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **31**, 703–716 (5 2012).

[16] Ghaida, R. S., Agarwal, K. B., Nassif, S. R., Yuan, X., Liebmann, L. W., and Gupta, P., "Layout decomposition and legalization for double-patterning technology," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **32**, 202–215 (2 2013).

[17] "Nangate Open Cell Library." http://www.si2.org/openeda.si2.org/projects/nangatelib.