INTEGRATION the VLSI journal 61 (2018) 77–87

Contents lists available at ScienceDirect

INTEGRATION, the VLSI journal

journal homepage: www.elsevier.com/locate/vlsi

Discrete relaxation method for contact layer decomposition of DSA with triple patterning☆

Xingquan Li, Jianli Chen, Wenxing Zhu*

Center for Discrete Mathematics and Theoretical Computer Science, Fuzhou University, Fuzhou, China

ARTICLE INFO

Keywords:
Mask and template assignment
Directed self-assembly
Triple patterning lithography
Conflict
Template cost
General layout

ABSTRACT

Block copolymer directed self-assembly (DSA) is a simple and promising candidate next-generation device fabrication technology, which is low-cost as complement with multi-patterning for contact layer patterning. In this paper, we consider the contact layer mask and template assignment problem of DSA with triple patterning lithography of general layout. To address this problem, first we construct a weighted conflict grouping graph, in which edges with negative weights are introduced, then a discrete relaxation based mask assignment problem is proposed. The integer linear program (ILP) formulation of the discrete relaxation problem is solved for obtaining a lower bound on the optimal value of this problem. In order to improve the lower bound, some valid inequalities are introduced to prune some poor relaxation solutions. At last, the obtained discrete relaxation solution is transformed to a legal solution of the original problem by solving a template assignment problem on the layout graph, which provides an upper bound on the optimal value of the original problem. Experimental results and comparisons show the effectiveness and efficiency of our method. In addition, under the discrete relaxation theory, the quality of our experimental results can be evaluated by the obtained upper and lower bounds. Specifically, the gap between the obtained upper and lower bounds is 0 for most of the sparse benchmarks, and the average gap is 0.4% for dense benchmarks.

1. Introductions

As the pitch size between features shrinking and the number of nodes increasing, manufacture of integrated circuit (IC) layout is more and more difficult. This urges on series of manufacture technologies, such as 193 nm ArF immersion optical lithography and the related multiple patterning lithography, electron beam lithography, block copolymer directed self-assembly, and extreme ultra violet lithography [1–3]. IC layouts consist of patterned lines and holes. The lines define the active device regions, gate electrodes, and the wirings between the devices. The holes define the electrical contacts between the wires and the transistors [4,5]. Some of the above manufacture technologies are popularly used to pattern line features in a layout [6], but the DSA technology is fit for patterning the dense hole features [7]. Especially, in 7 nm nodes distribution of the features on contact/via layer is dense and aligned [8], hence the DSA technique is necessary.

To pattern contact holes by DSA, guiding templates are usually used to form contacts [9,10]. For sparse structure, a number of single-hole templates are used to form contacts. For dense structure, too close templates would generate conflicts [11–13]. To reduce the conflicts, some of the contacts within a short distance would be grouped together in a multi-hole template [11–13]. As shown in Fig. 1(a), the left contact is contained in a single-hole template, and the right two close contacts are grouped in a two-hole template.

However, grouping more than one contacts in a multi-hole template may introduce overlays. For different guiding templates with different shapes or sizes, the overlays are different. Specifically, complex (irregular shape) guiding templates may introduce large overlays and the contained contacts may not be patterned correctly [11]. Hence, during template assignment, the cost of a guiding template should be considered.

Furthermore, for a very dense contact layer layout, the contact layer fabricated by single patterning is unqualified due to a number of conflict errors. Hence the DSA with multiple patterning (DSA-MP) technology is a solid choice, and a crucial problem in DSA-MP is the mask and template assignment. An example of mask and template assignment for DSA with triple patterning (DSA-TP) is shown in Figs. 1(b) and (c). Fig. 1(c) is a template assignment of the layout in Fig. 1(b), where the three colors represent three masks, and the right two contacts are contained in a vertical two-hole template, and a

☆ Research supported by the National Natural Science Foundation of China under Grants 61672005 and 11501115.

* Corresponding author.
E-mail address: wxzhu@fzu.edu.cn (W. Zhu).

https://doi.org/10.1016/j.vlsi.2017.11.004
Received 14 April 2017; Received in revised form 3 September 2017; Accepted 16 November 2017
Available online 06 December 2017
0167-9260/ © 2017 Elsevier B.V. All rights reserved.

Fig. 1. An example of contact layer decomposition for DSA with TPL. (a) A template assignment for the sparse layout. (b) A dense layout. (c) A mask and template assignment for the dense layout.

conflict is generated between the two green one-hole templates due to the small pitch between them.

Recently, some works concerned the mask and template assignment problem of DSA-MP [12,13], including the mask and template assignment problem of DSA with double patterning (MTADD) and with triple patterning (MTADT). For the MTADD problem, Ref. [13] has obtained good enough solutions for the tested benchmarks comparing with the solutions of the exact integer linear program formulation. However, the solutions still have many unresolved conflicts, although under their conflict spacing setting, the distributions of contacts in the tested layouts are sparse. Therefore, it is necessary to use triple masks for the contact layer with 7 nm nodes. In this paper, we consider the mask and template assignment problem of DSA-TP (MTADT).

For the row structure layout, Xiao et al. [11] proposed three methods and compared their effectiveness. These methods are: 1) color first iterative; 2) group first iterative; 3) shortest path based optimal decomposition. By comparisons, the shortest path based method achieved the best decomposition. For the general layout, Badr et al. [12] first considered the MTADM problem and formulated it as an integer linear programming problem, and proposed a maximum cardinality matching (MCM) based method to quickly obtain a result. However, the method in [12] has some issues. In the aspect of problem formulation, the method in [12] does not consider the template cost, which is different for different types of templates. In the aspect of solution method, Ref. [12] proposed two methods for the MTADT problem. However, since many variables and constraints of template grouping are introduced, the ILP formulation is too complex to fast solve. Moreover, the MCM based method is a grouping first method, and the solution quality is unknown.

In order to improve the quality of decomposition results of the MTADT problem, Kuang et al. [13] considered the simultaneous template optimization and mask assignment problem of DSA with triple patterning. They proposed a look-up table (LUT) based assignment method, which finds all the possible 3-colorable sub-graphs by removing some edges. The method is fast and effective for sparse and small graph. However, since the number of 3-colorable sub-graphs of a graph is exponential, the storage size of LUT would not be scalable for very dense or large graph, and it is time consuming to check the LUT. In order to reduce runtime, the method in [13] does not store and check all 3-colorable sub-graphs. This will lose optimality of the results, and the gap between an obtained solution and the optimal solution is still unknown.

In this paper, we propose a discrete relaxation based decomposition method to solve the MTADT problem of general layout. The discrete relaxation method is a general scheme for dealing with hard discrete optimization problems, which relaxes a hard problem to an easier one, and then the relaxation solution is legalized to a solution of the initial problem [14]. An advantage of the method is that the solution quality can be evaluated in the experiment. The evaluation of a solution is significant for an NP-hard problem. If we know the gap between an obtained result and the optimal value of an instance, then we will know whether the solution is good or not. This scheme has been proposed and used to address the triple patterning layout decomposition problem [14]. However, the discrete relaxation method should be designed carefully according to the feature of an addressed problem.

For the MTADT problem of general layout, our main contributions are listed as follows.

- We sum up general rules for the costs of vertical or horizontal templates with different sizes, and construct a weighted conflict grouping graph.
- Basing on the weighted conflict grouping graph, we propose a novel integer linear program for the MTADT problem, which is not equivalent to the MTADT problem but provides a lower bound on the optimal value of the MTADT problem. Moreover, some valid inequalities are introduced for cutting some no good solutions, and obtaining a better lower bound.
- We propose a template assignment approach to transform a relaxation solution to a feasible solution of the MTADT problem, which provides an upper bound on the optimal value of the MTADT problem. According to the obtained lower bound and upper bound, we can evaluate the quality of our experimental results. Specially, if the upper bound is equal to the lower bound, then we obtain an optimal solution of the MTADT problem.
- Comparisons of experimental results show that our decomposition method is effective. More specifically, the gap between the obtained upper and lower bounds is 0.0% for most of the sparse benchmarks, which shows the optimality of the obtained results. And the average gap is 0.4% for the dense benchmarks, which shows the goodness of the obtained results for dense layouts.

The rest of this paper is organized as follows. Section 2 shows the template types and problem formulation. Section 3 introduces the discrete relaxation decomposition method, and the feasible solution generation method is introduced in Section 4. Experimental results are presented in Section 5, and conclusions of our work are made in Section 6.

2. Preliminaries

In this section, first we introduce the types of the DSA guiding templates, and then we describe the mask and template assignment problem of DSA-TP.

2.1. DSA guiding template

To print contact holes by DSA, guiding templates are needed, which are usually fabricated by conventional optical lithography technology [5]. Thus the resolution is limited by the pitch of guiding templates. For sparse structure, the contact pitch is big enough, hence the contacts can be contained in a series of single-hole templates. But for dense structure, the contact pitch is too small to satisfy the resolution for numerous single-hole templates, and multi-hole template would be used to guide a group of contacts for improving the resolution.

Theoretically, the type of multi-hole template could be of any shape [15]. However, complex guiding template may introduce large overlay and the intended contacts may not be patterned correctly [11]. Such as the diagonal templates (Fig. 2(e)), the local diagonal templates (Fig. 2(f)), or the “L” shape templates (Fig. 2(g)), they cannot be printed reliably, hence the results after printed should be verified by

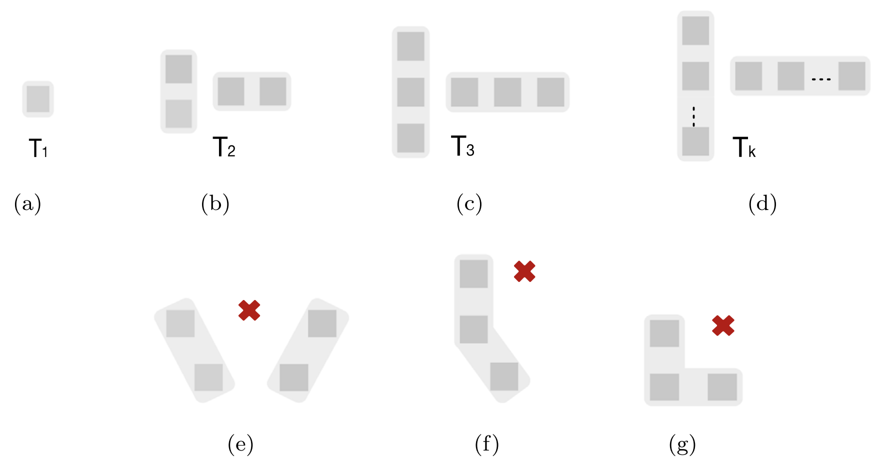

Fig. 2. Template types. (a)-(d) Available vertical and horizontal templates. (e)-(g) Illegal templates.

the optical proximity correction process, which is of high cost [13,16]. Furthermore, in order to consider the complex templates for the DSA technology, the grid model should be modeled for the contact layer layout [17,18]. However, for some special contact layer layout, under the given conflict spacing and grouping spacing, some contacts do not align to the gird line. This may lead to that some templates cannot properly guide the matching contacts. In this paper, we consider the MTADT problem without grid model. Hence under the gridless assumption and for the reason of avoiding high correction cost, we only consider the vertical and horizontal templates as in [11,13]. That is, only a group of contacts in a vertical or horizontal line can be grouped into a template.

For vertical or horizontal templates, different sizes of templates have different costs [5]. The main factors deciding the cost of a template are the number of holes in the template and the size of the template. It must be remarked that, the hole pitches in a template may not be uniform since the distribution of contacts in a layout may not be regular. Thus, for two templates with the same number of holes, their costs may be different. But for simplicity, we only consider in this paper the number of holes as the evaluation of cost of a vertical or horizontal template as in [13].

We sum up three rules on the cost of a template, which are important but not unique:

1. A template with more holes will have higher cost.
2. A template will have higher cost than two or more templates for grouping the same number of holes.
3. Suppose that any two neighboring holes in a template is regarded as a pair. A template will have less cost than two or more templates if the latter templates have the same number of pairs as the former one.

Rule 1) is due to that, a template with more holes is more difficult to control the lithographic variations [15,16]. Rule 2) is due to that, a template containing several holes is more difficult for lithographic variations than several other templates containing these holes. As for rule 3), the latter templates involve more contacts, which may generate more manufacture errors [16]. Figs. 3(a)-(c) show examples for rules 1–3, respectively.

Let T_k be a template with k holes. Figs. 2(a)-(d) show the vertical and horizontal templates, $T_1, T_2, T_3, \cdots$, and T_K, respectively, where K is the maximum number of holes in a template. Let $cost_{T_k}$ be the cost of template T_k, and we suppose that $cost_{T_k}$ is an integer in this paper. According to the above three rules, the costs of templates are set as: i) $cost_{T_1} = 0$. This is because every contact can be guided by a single-hole template, while the use of multi-hole templates is for eliminating conflicts which needs extra cost; ii) $cost_{T_2} = 3$, as a baseline; iii) $2cost_{T_{k-1}} \geq cost_{T_k} > \frac{k}{k-1} cost_{T_{k-1}}$, $k = 3, 4, \cdots K$. This inequality indicates that the average cost of holes in T_k should be greater than that of holes in T_{k-1}.

The above setting rules for the cost of template is compatible with the settings in previous works [11,13,16]. In [16], Xiao et al. formulated an equation for calculating the cost of template: $c_i = \lambda \times p_i$, where c_i denotes the cost of the i^{th} multiple template, and p_i is the number of templates pairs in the multiple template, i.e., $p_i = k - 1$ for template T_k. Suppose the cost of template T_2 is $cost_{T_2}$, then $cost_{T_3} = 2cost_{T_2}$, $cost_{T_4} = 3cost_{T_2} = \frac{3}{2} cost_{T_3}$, $\cdots$. It is easy to show that $2cost_{T_{k-1}} \geq cost_{T_k} > \frac{k}{k-1} cost_{T_{k-1}}$ includes the above equalities. Furthermore, as an example of setting in [11], Xiao et al. set the costs of templates $cost_{T_1}$, $cost_{T_2}$, and $cost_{T_3}$ as 0, 5, 8, respectively. This setting still satisfies $2cost_{T_{k-1}} \geq cost_{T_k} > \frac{k}{k-1} cost_{T_{k-1}}$. In another work [13], Kuang et al. assumed that the cost of a template with more than 2 holes is always larger than the summation of the costs of the constituent templates, e.g., $cost_{T_3} > cost_{T_1} + cost_{T_2}$ and $cost_{T_4} > 2cost_{T2}$. This assumption is also compatible with our assumption.

Note that, when $cost_{T_3} = 3$ and $cost_{T_k}$ is an integer, it is easy to show that $cost_{T_k} = 2k - 1$ is the tightest setting for satisfying $2cost_{T_{k-1}} \geq cost_{T_k} > \frac{k}{k-1} cost_{T_{k-1}}$, $k = 3, 4, \cdots K$. That is, when $cost_{T_2} = 3$ and $cost_{T_k}$ is an integer, it holds that $cost_{T_k} \geq 2k - 1$.

2.2. Problem formulation

Some involved notations are introduced as follows:

- d_c, the minimum conflict spacing;
- $d_{g_{min}}$, the minimum grouping spacing;
- $d_{g_{max}}$, the maximum grouping spacing;
- C_1, C_2, C_3, the colors of TPL;
- β, the weighting parameter between the conflict number and the total cost of templates, which is set as $\beta = 0.01$.

For the above notations, $d_{g_{min}} < d_{g_{max}} < d_c$. If the distance between two contacts is less than the minimum conflict spacing d_c, and the two contacts are assigned to the same mask without grouping, then a conflict is generated between the two contacts. In order to reduce the number of conflicts, we group some contacts together according to the template types defined in Section 2.1. Then the MTADT problem is defined as follows:

The mask and template assignment problem of DSA-TP P_0.

Given: Contact layer layout, the set of vertical and horizontal templates, three masks, parameter β.

Find: A mask assignment for all contacts, and groups of some of the contacts by available multi-hole templates.

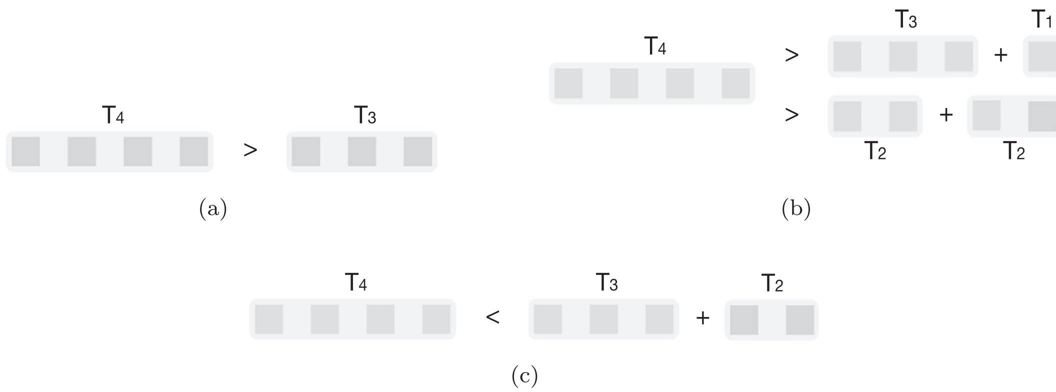

Fig. 3. Comparison of the costs of different templates. (a) Rule 1). (b) Rule 2). (c) Rule 3).

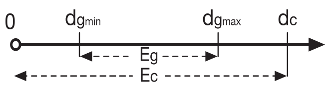

Fig. 4. Conflict spacing and grouping spacing.

Subject to: Every contact is assigned to only one of the three masks, and is assigned to only one of the templates. Moreover, all contacts in a template must be assigned to the same mask.

Objective: $|C| + \beta \cdot TCost$ is minimized, where $c_{ij} \in C$ denotes the conflict between contacts i and j, and $|C|$ is the number of conflicts, and $TCost$ is the total cost of used templates.

3. Discrete relaxation method for mask and template assignment of DSA with TPL

In this section, we construct the conflict grouping graph (CGG) for a layout, and propose a discrete relaxation of the MTADT problem using an ILP formulation. In order to obtain a better relaxation solution, we introduce some valid inequalities for the ILP problem.

Before showing the discrete relaxation method for the MTADT problem, we introduce the definition of discrete relaxation and its propositions as follows.

Definition 1 (*Discrete relaxation* [14]). Problem RP: $z^R = \min\{f^R(x): x \in X^R\}$ is a discrete relaxation of problem P: $z = \min\{f(x): x \in X\}$, if there exists an optimal solution $x^{R\star}$ of problem RP, and there exists an optimal solution $x^\star$ of problem P such that $f^R(x^{R\star}) \le f(x^\star)$.

Proposition 1 ([14]). *If problem RP is a discrete relaxation of problem P, then* $z^R \le z$.

Proposition 1 means that, we will obtain a lower bound on the minimum value of the original problem by solving the discrete relaxation problem. Specially, we have

Proposition 2 ([14]). *Suppose that problem RP is a discrete relaxation of problem P. Let $x^{R\star}$ be an optimal solution of problem RP. If $x^{R\star}$ can be transformed to a feasible solution x of problem P, such that $f^R(x^{R\star}) = f(x)$, then x is an optimal solution of problem P.*

For the discrete relaxation method, the function $f^R(x)$ and the solution set X^R must be carefully selected. Generally, we should select an X^R such that an optimal solution of problem RP can be transformed easily to a feasible solution of problem P, and the gap between the minimum values of problems P and RP is not too large.

3.1. Conflict grouping graph construction

First, we define the conflict grouping graph as follows.

Definition 2 (*Conflict grouping graph CGG*). The conflict grouping graph is defined as an undirected graph $CGG(V, E_c)$, where V is the set of vertices, E_c is the set of conflict edges.

If the distance between two contacts i and $j \in V$ is less than d_c, then there exists a conflict edge $e_{ij} \in E_c$ between them; if the distance between two contacts i and $j \in V$ is between $d_{g_{min}}$ and $d_{g_{max}}$, and i and j are in the vertical or horizontal line, then there exists a grouping edge $e_{ij} \in E_g$ between them. Obviously, $E_g \subseteq E_c$.

According to the distances between contacts, a layout with contacts is transformed to a conflict grouping graph. This can be achieved in $O(kn)$ runtime, where n is the number of contacts, and k is the maximum number of contacts within the minimum conflict spacing d_c of contacts. Fig. 4 illustrates the conflict spacing and grouping spacing, and an example of conflict grouping graph construction is shown as Fig. 5(b), where all lines are the conflict edges and dotted lines are the grouping edges.

3.2. Discrete relaxation based mask assignment

Discrete relaxation is an optimization method, which relaxes a hard minimization problem to an easier one by some relaxation techniques [14]. An optimal solution of the relaxation problem provides a lower bound on the minimal value of the original problem. In this section, we propose a way of discrete relaxation for the MTADT problem.

3.2.1. Weighted conflict grouping graph construction

In the conflict grouping graph CGG, two contacts connected by a conflict edge should be assigned to different masks or grouped by a template for MTADT. In order to reduce the conflicts and the total cost

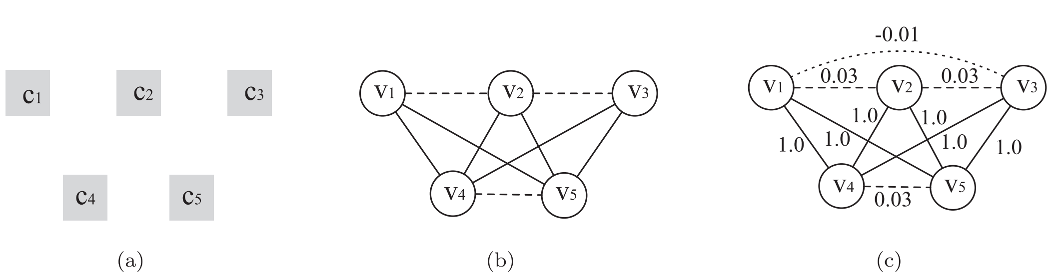

Fig. 5. Conflict grouping graph construction.

of used templates, we need to decide which contacts should be grouped first, and which conflict edge could not be eliminated by grouping. We distinguish the conflict edges by weighting them, and then construct the weighted conflict grouping graph (WCGG). In order to handle the grouping, we introduce a definition of negative edge as follows.

Definition 3 (*Negative edge ne*). A negative edge is an undirected edge with negative weight in a graph. If there exist two vertices i and j connected to the same vertex k by grouping edges, i.e., $ge_{ik} \in E_g$, $ge_{jk} \in E_g$, and contacts i, j are in the same vertical or horizontal line, then e_{ij} is added to the graph and called a negative edge.

Let E_n be the set of negative edges. We define the *WCGG* as follows:

Definition 4 (*Weighted conflict grouping graph WCGG*). The weighted conflict grouping graph is an undirected edge-weighted graph $WCGG(V, E, W)$, where V is the set of vertices, E is the set of edges, $E = E_c \cup E_n$, and W is the set of weights of edges in E.

The weighting rule for edge $e_{ij} \in E$ between contacts i and j is set as

$$w_{ij} = \begin{cases} 1.0, & \text{if } e_{ij} \in E_c - E_g; \\ 0.03, & \text{if } e_{ij} \in E_g; \\ -0.01, & \text{if } e_{ij} \in E_n. \end{cases}$$

Fig. 5(c) shows an example of weighted conflict grouping graph.

According to the above weighting rule, we can see that:

1. If contacts i_1, i_2 are assigned to the same mask, and $e_{i_1i_2} \in E_c - E_g$, then there exists a conflict between i and j, and the edge cost is 1.0;
2. If contacts i_1, i_2 are assigned to the same mask, and $e_{i_1i_2} \in E_g$, then the edge cost is 0.03;
3. If contacts i_1, i_2, i_3 are assigned to the same mask, and $e_{i_1i_2} \in E_g$, $e_{i_2i_3} \in E_g$, $e_{i_1i_3} \in E_n$, then the edge cost is 0.05;
4. If contacts $i_1, i_2, \cdots, i_k$, $k = 3, 4$, are assigned to the same mask, and $i_1, i_2, \cdots, i_k$ satisfy the T_k template condition. It can be deduced that there are $k-1$ grouping edges and $k-2$ negative edges, and then the edge cost is $0.03(k-1) - 0.01(k-2) = 0.01(2k-1)$; Figs. 6(a)-(c) and Fig. 6(e) show examples for the above cases 1–4.

3.2.2. Discrete relaxation

We consider the following problem P_1 on the weighted conflict grouping graph:

$$\min \sum_{e_{ij} \in E} w_{ij}(x_i == x_j) \tag{1}$$

$$\text{s. t.} \quad x_i \in \{1, 2, 3\}, \quad \forall\, i \in V, \tag{1a}$$

where x_i denotes the assigned mask of vertex i.

Suppose X_0 and X_1 are the solution spaces of problems P_0 and P_1, respectively. For any solution x^0 of problem P_0, every contact has been assigned to a mask and a template. However, for any solution x^1 of problem P_1, every contact has been assigned to a mask, but has not been assigned to a template. That means, for any $x^0 \in X_0$, we can get a solution $x^1 \in X_1$ from x^0 by omitting the template assignment.

Lemma 1. *Suppose $x^0 \in X_0$ is transformed to $x^1 \in X_1$ by omitting the template assignment, X_0 and X_1 are the solution spaces of problems P_0 and P_1, respectively, and $f_0(x^0)$ and $f_1(x^1)$ are the objective functions of problems P_0 and P_1, respectively. Then $f_1(x^1) \le f_0(x^0)$.*

Proof. For any $x^0 \in X_0$, the total cost of problem P_0 is

$$f_0(x^0) = |C| + \beta \cdot TCost = |C| + 0.01 \sum_{k=2}^{K} cost_{T_k}|T_k|,$$

where K is the number of template types, $|C|$ is the number of total conflicts. If contacts i and j are assigned to the same mask, $e_{ij} \in E_c$, and i and j are not assigned to the same template, then a conflict is generated between i and j, i.e., $c_{ij} = 1$. $|T_k|$ is the number of used k-hole templates in x^0, where the contacts in the same k-hole template should be in the same mask, and satisfy the vertical or horizontal k-hole template conditions.

The cost of x^0 consists of conflict cost and template cost. x^1 is obtained from x^0 by omitting the template assignment. Let $E = E_T \cup E_D$, where E_T is the set of edges between contacts i and j in the same template of x^0. E_D is the set of edges between contacts i and j in different templates of x^0. Then

$$\begin{aligned} f_1(x^1) &= \sum_{e_{ij} \in E} w_{ij}(x_i^1 == x_j^1) \\ &= \sum_{e_{ij} \in E_D} w_{ij}(x_i^1 == x_j^1) + \sum_{e_{ij} \in E_T} w_{ij}(x_i^1 == x_j^1). \end{aligned}$$

First, for the contacts i and j that are in the same template of x^0, according to the weight setting of problem P_1, it holds that

$$\begin{aligned} \sum_{e_{ij} \in E_T} w_{ij}(x_i^1 == x_j^1) &= \sum_{e_{ij} \in E_T} w_{ij} \\ &= \sum_{k=2}^{K} \{[0.03(k-1) - 0.01(k-2)]|T_k|\} \\ &= 0.01 \sum_{k=2}^{K} (2k-1)|T_k| \\ &\le 0.01 \sum_{k=2}^{K} cost_{T_k}|T_k|. \end{aligned}$$

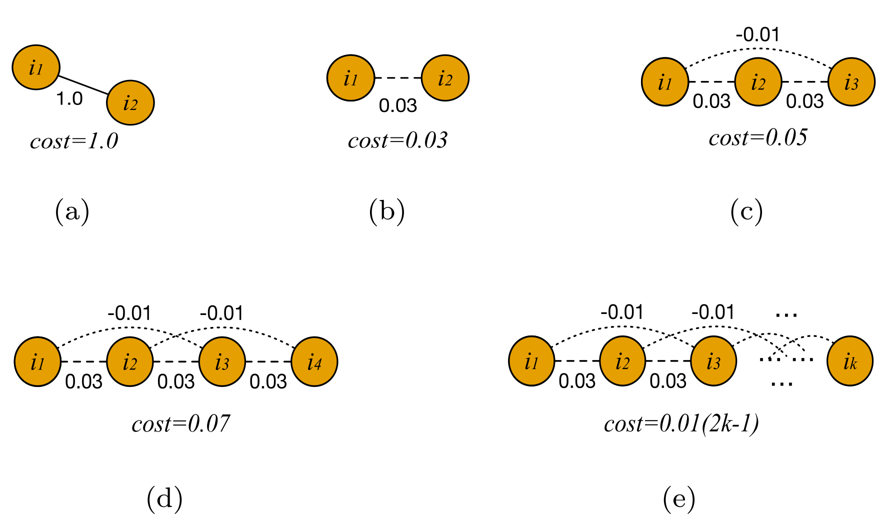

Fig. 6. Total edge costs of different templates.

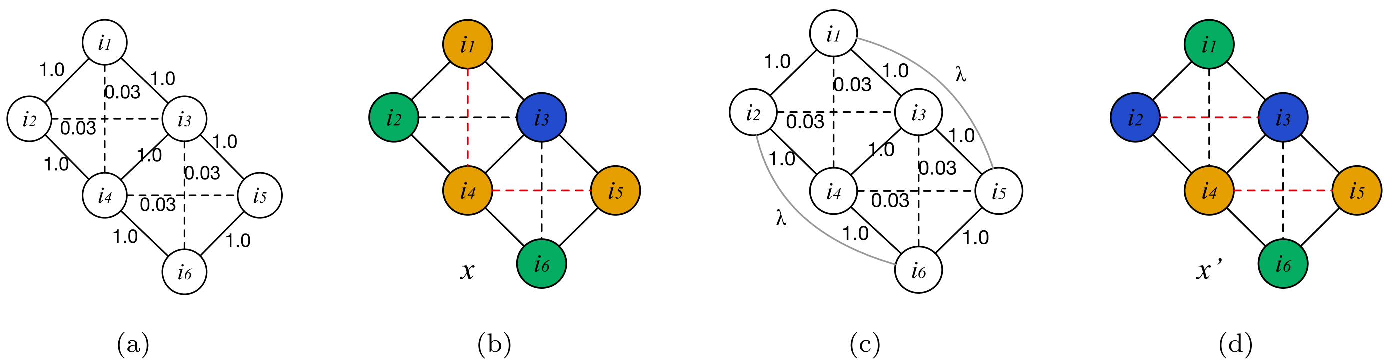

Fig. 7. Corner incompatibility and triangle edges. (a) Weighted conflict grouping graph. (b) A solution obtained by solving problem P_2. (c) New weighted conflict grouping graph. (d) A solution obtained by solving problem P_2^+.

The last inequality is due to $cost_{T_k} \geq 2k-1$ for $k \geq 2$.

Second, we consider the cost between the contacts i and j that are in different templates of x^0. Here, we only consider the cost between contacts i and j that are in the same mask, since if i and j are in different masks, then the cost is 0. Let E'_D be the set of edges $e_{ij} \in E_D$ and i and j are in the same mask. Let $E_{D_1} = E'_D \cap (E_c - E_g)$, $E_{D_2} = E'_D \cap E_g$, and $E_{D_3} = E'_D \cap E_n$. Then $E'_D = E_{D_1} \cup E_{D_2} \cup E_{D_3}$. Moreover, $|E'_D \cap E_c| = |E_{D_1} \cup E_{D_2}| = |E_{D_1}| + |E_{D_2}|$ which is the number of conflicts $|C|$. Then

$$\begin{aligned}\sum_{e_{ij}\in E_D} w_{ij}(x_i^1 == x_j^1) &= \sum_{e_{ij}\in E'_D} w_{ij} \\ &= \sum_{e_{ij}\in E_{D_1}} w_{ij} + \sum_{e_{ij}\in E_{D_2}} w_{ij} + \sum_{e_{ij}\in E_{D_3}} w_{ij} \\ &= |E_{D_1}| + 0.03|E_{D_2}| - 0.01|E_{D_3}| \\ &\leq |E_{D_1}| + |E_{D_2}| = |C|.\end{aligned}$$

Therefore, for any $x^0 \in X_0$, and for $x^1 \in X_1$ which is obtained from x^0 by omitting the template assignment, we have

$$\begin{aligned}f_1(x^1) &= \sum_{e_{ij}\in E} w_{ij}(x_i^1 == x_j^1) \leq |C| + 0.01 \sum_{k=2}^{K} cost_{T_k}|T_k| \\ &= f_0(x^0).\end{aligned}$$

□

Theorem 1. *Problem P_1 is a discrete relaxation of problem P_0.*

Proof. Suppose x^{0*} and x^{1*} are optimal solutions of problems P_0 and P_1, respectively. By Lemma 1, we have

$$f_1(x^{1*}) \leq f_1(x^{1'}) \leq f_0(x^{0*}),$$

where $x^{1'}$ is obtained from x^{0*} by omitting the template assignment. Hence, problem P_1 is a discrete relaxation of problem P_0. □

Corollary 1. *Suppose an optimal solution $x^{1*} \in X_1$ of problem P_1 is transformed to $x^{0*} \in X_0$ by some template assignment. If $f_0(x^{0*}) = f_1(x^{1*})$, then x^{0*} is an optimal solution of problem P_0.*

Corollary 1 holds obviously from Theorem 1. By Theorem 1, the optimal value LB of problem P_1 is a lower bound on the optimal value OPT of problem P_0. By legalizing an optimal solution of problem P_1 to a feasible solution of problem P_0, we can obtain an upper bound UB on the optimal value OPT of problem P_0, and it holds $UB - OPT \leq UB - LB$. Thus Theorem 1 can be used to evaluate the quality of our experimental results, i.e., the gap between our experimental result and the optimal value of problem P_0.

In order to solve the discrete relaxation problem P_1, we transform P_1 to an Integer Linear Programming (ILP) problem P_2 equivalently as follows:

$$\min \sum_{e_{ij}\in E} w_{ij} c_{ij} \tag{2}$$

$$\text{s. t. } x_{im} + x_{jm} \leq 1 + c_{ij}, \quad \forall\, e_{ij} \in E,\ m = 1, 2, 3; \tag{2a}$$

$$\sum_{m=1}^{3} x_{im} = 1, \quad \forall\, i \in V; \tag{2b}$$

$$x_{im},\ c_{ij} \in \{0, 1\}, \quad \forall\, i \in V, \forall\, e_{ij} \in E,\ m = 1, 2, 3. \tag{2c}$$

In the above formulation, x_{im} is a binary variable, which denotes the assigned mask for contact i. If $x_{im} = 1$, then i is assigned to mask m. c_{ij} is a binary variable, which is used for indicating whether a conflict is generated between contacts i and j.

In problem P_2, constraints (2a) are used to decide whether a conflict c_{ij} is generated between contacts i and j. That is, for $e_{ij} \in E$, if contacts i and j are in the same mask, then $c_{ij} = 1$. Constraint (2b) is used to select one of the three masks for contact i.

3.3. Improving the lower bound by adding valid inequalities

The discrete relaxation problem P_2 can be solved for obtaining a lower bound of problem P_0. However, the gap between the lower bound and the optimal value of problem P_0 may be large, and the obtained decomposition result might be of poor quality. The proposed discrete relaxation method is not only for finding a lower bound of the MTADT problem, but also for obtaining a good solution of the MTADT problem. Hence in this subsection we improve the lower bound provided by problem P_2 by adding some valid inequalities

In the weighted conflict grouping graph $WCGG$ of Fig. 7(a), the solid lines are conflict edges, and their weights are $w = 1.0$ respectively; while the dotted lines are grouping edges, and their weights are $w = 0.03$ respectively. Fig. 7(b) shows an optimal solution x of problem P_2, which has two conflict variables $c_{i_1 i_4}$ and $c_{i_4 i_5}$ equal to 1, and the objective value of problem P_2 is 0.06. However, contacts i_1, i_4 and i_5 cannot be assigned to the same template at the template assignment stage. Only one of the two conflict grouping edges $e_{i_1 i_4}$ and $e_{i_4 i_5}$ can be grouped by a T_2 template, another one would cause a conflict, and the total cost of P_0 is $0.01 \times cost_{T_2} + |C| = 1 + 0.01 \times cost_{T_2}$. Thus, the gap between the objective values of the discrete relaxation problems P_2 and the problem P_0 is $0.94 + 0.01 \times cost_{T_2}$. In the following, we let $\lambda = 0.94 + 0.01 \times cost_{T_2}$.

In order to reduce the gap, and further improve the quality of an obtained solution of problem P_0, we should handle the case as in Fig. 7(b). We call the case as corner incompatibility (CI). More formally, corner incompatibility is that, there exist at least two grouping edges incident to a contact k called corner contact, and at least two grouping edges containing k are orthogonal. Since corner incompatibility is not allowed, we preclude in this section corner incompatibility by introducing some valid inequalities. The technique is adding some extra edges, and these edges are called triangle edges, which is defined as:

Definition 5 (*Triangle edge te*). A triangle edge is an undirected edge in a graph. If there exist two vertices i and j connected to the same

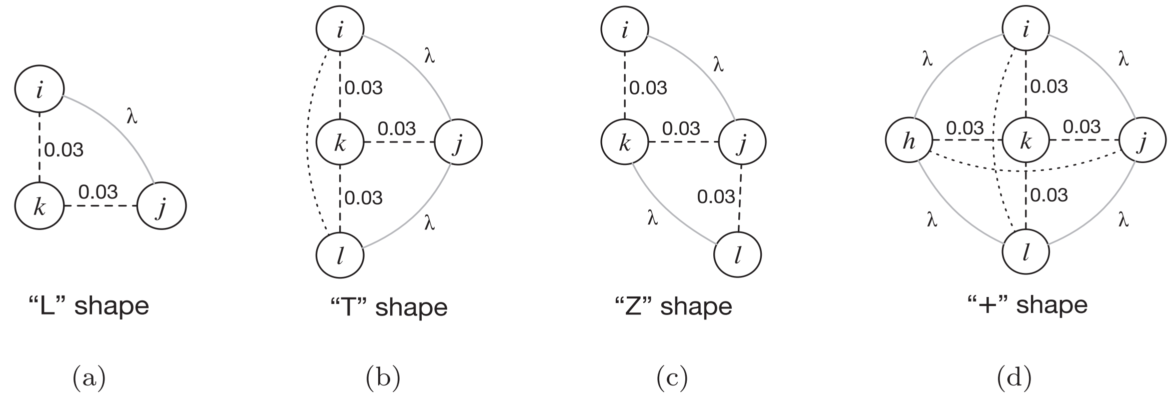

Fig. 8. Structures with corner incompatibility. (a) "L" shape. (b) "T" shape. (c) "Z" shape. (d) "+" shape.

vertex k by grouping edges, i.e., $e_{ik} \in E_g$, $e_{jk} \in E_g$, and if $e_{ij} \notin E_c \cup E_n$, then e_{ij} is added to the graph and is called a triangle edge. Let E_t be the set of triangle edges.

In Fig. 8, there are four types of structures with corner incompatibility structure (*CIS*): i) "L" shape, $\{e_{ik}, e_{jk}\} \subseteq E_g$ and $\{e_{ij}\} \subseteq E_t$ as shown in Fig. 8(a); ii) "T" shape, $\{e_{ik}, e_{jk}, e_{lk}\} \subseteq E_g$, $\{e_{il}\} \subseteq E_n$ and $\{e_{ij}, e_{jl}\} \subseteq E_t$ as shown in Fig. 8(b); iii) "Z" shape, $\{e_{ik}, e_{jk}, e_{jl}\} \subseteq E_g$, and $\{e_{ij}, e_{kl}\} \subseteq E_t$ as shown in Fig. 8(c); and iv) "+" shape, $\{e_{ik}, e_{jk}, e_{lk}, e_{hk}\} \subseteq E_g$, $\{e_{il}, e_{hl}\} \subseteq E_n$ and $\{e_{ij}, e_{jl}, e_{lh}, e_{hi}\} \subseteq E_t$ as shown in Fig. 8(d).

In order to avoid the contacts in a *CIS* being assigned to the same mask, we modify the *WCGG* as the *newWCGG* by adding some triangle edges, and assign the weight of every triangle edge as $w = \lambda$. Then we add four kinds of new constraints to problem P_2, and call the new problem as P_2^+.

For the "L" shape structure, if the three contacts i, j and k are assigned to the same mask, then there exists at least a conflict due to corner incompatibility. In order to obtain a better solution, we add some new constraints for contacts i, j and k of the "L" shape structure to avoid the three contacts i, j and k being assigned to the same mask. These constraints are

$$x_{im} + x_{km} + x_{jm} \le 2 + c_{ij}, \quad m = 1, 2, 3, \tag{3a}$$

where contact k is a corner contact. Constraints (3a) are used to restrict that, if the contacts i, j and k are assigned to the same mask, then the conflict variable of triangle edge $e_{ij} \in E_t$ has $c_{ij} = 1$.

For the "T" shape structure, if the four contacts i, j, l and k are assigned to the same mask, then there exists at least a conflict due to corner incompatibility. Similarly, we add some valid inequalities for the four contacts of the "T" shape structure. These constraints are

$$x_{im} + x_{km} + x_{jm} \le 2 + c_{ij} + c_{jl}, \quad m = 1, 2, 3; \tag{4a}$$

$$x_{jm} + x_{km} + x_{lm} \le 2 + c_{ij} + c_{jl}, \quad m = 1, 2, 3, \tag{4b}$$

where contact k is a corner contact. Constraints 4(a)–4(b) are used to restrict that, if the three contacts i, j and k (or j, l and k) or the four contacts are assigned to the same mask, then at least one of the conflict variables of the triangle edges e_{ij} and e_{jl} is equal to 1.

For the "Z" shape structure, if the four contacts i, j, l and k are assigned to the same mask, then there exists at least a conflict due to corner incompatibility. Similarly, we add some valid inequalities for the four contacts of the "Z" shape structure. These constraints are

$$x_{im} + x_{km} + x_{jm} \le 2 + c_{ij} + c_{kl}, \quad m = 1, 2, 3; \tag{5a}$$

$$x_{jm} + x_{km} + x_{lm} \le 2 + c_{ij} + c_{kl}, \quad m = 1, 2, 3, \tag{5b}$$

where contacts k and j are corner contact. Constraints 5(a)–5(b) are used to restrict that, if the three contacts i, j and k (or j, l and k) or the four contacts are assigned to the same mask, then at least one of the conflict variables of the triangle edges e_{ij} and e_{kl} is equal to 1.

For the "+" shape structure, if the five contacts i, j, l, h and k are assigned to the same mask, then there exists at least two conflicts due to corner incompatibility. Similarly, we add some valid inequalities for the five contacts of the "+" shape structure. These constraints are

$$x_{im} + x_{km} + x_{jm} \le 2 + c_{ij} + c_{jl}, \quad m = 1, 2, 3; \tag{6a}$$

$$x_{jm} + x_{km} + x_{lm} \le 2 + c_{jl} + c_{lh}, \quad m = 1, 2, 3; \tag{6b}$$

$$x_{lm} + x_{km} + x_{hm} \le 2 + c_{lh} + c_{hi}, \quad m = 1, 2, 3; \tag{6c}$$

$$x_{hm} + x_{km} + x_{im} \le 2 + c_{hi} + c_{ij}, \quad m = 1, 2, 3, \tag{6d}$$

where contact k is a corner contact. Constraints 6(a)–6(d) are used to restrict that, if the three contacts in an "L" shape *CIS* or the four contacts in a "T" shape *CIS* are assigned to the same mask, then at least one of the conflict variables of the triangle edges is equal to 1. If the five contacts in a "+" shape *CIS* are assigned to the same mask, then two conflict variables among the triangle edges e_{ij}, e_{jl}, e_{lh} and e_{hi} are equal to 1.

Actually, according to our experiments, the number of "T" shapes, "Z" shapes and "+" shapes *CIS* is very small. After adding the *CIS* constraints, the *newWCGG* of the structure in Fig. 7(b) is constructed as Fig. 7(c). Then by solving problem P_2^+, we can obtain a discrete relaxation solution as Fig. 7(d) instead of Fig. 7(b).

Theorem 2. *Problem P_2^+ is still a discrete relaxation of problem P_0.*

Proof. Suppose that f_2^+, f_2 and f_0 are the objective functions of problems P_2^+, P_2 and P_0, respectively. Suppose x^0 is an optimal solution of problem P_0, and x is obtained from x^0 by omitting the template assignment, then x is a solution of both problems P_2^+ and P_2, respectively. The difference between problems P_2^+ and P_2 is the triangle edges of the three types of *CIS*. We analyze the cost of the three types of *CIS* in problem P_2^+.

Suppose S_L is an "L" shape *CIS*, i, j and k are the three contacts of S_L, k is a corner contact and te_{ij} is the triangle edge. Let $x_L = \{x_{i1}, x_{i2}, x_{i3}, x_{j1}, x_{j2}, x_{j3}, x_{k1}, x_{k2}, x_{k3}\}$. Apparently, x_L is a part of x. Let x_L^0 be the part of the solution x^0 of problem P_0 for S_L. If i, j are assigned to different masks, then the conflict variable of the triangle edge te_{ij} has $c_{ij} = 0$ for minimizing the objective. If the mask of contact k is different from both of the masks of i and j, then $f_2^+(x_L) = f_0(x_L^0) = 0$; if the mask of k is the same as one of the masks of i and j, say i, then $f_2^+(x_L) = 0.03$. And for x_L^0, if the grouping edge ge_{ki} is guided by a multi-hole template, then $f_0(x_L^0) = 0.01 \times cost_{T_2} \ge 0.03$; otherwise a conflict is generated between k and i, and $f_0(x_L^0) = 1$. Anyway, it holds that $f_2^+(x_L) \le f_0(x_L^0)$.

If all contacts i, j and k of S_L are assigned to the same mask, then constraint (3a) will force c_{ij} to be 1. On the one hand, $f_2^+(x_L) = 0.06 + \lambda = 1 + 0.01 \times cost_{T_2}$. On the other hand, since S_L is an "L" shape *CIS*, (1) if in x_L^0, S_L needs a conflict and a multi-hole template to contain one of the grouping edges in S_L, then the decomposition cost for S_L is $f_0(x_L^0) = |C| + TCost = 1 + 0.01 \times cost_{T_2}$;

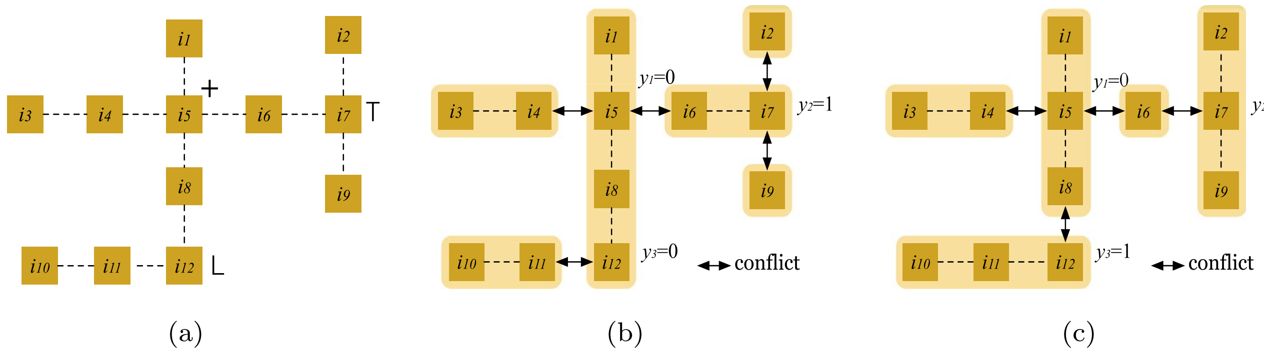

Fig. 9. Template assignment for layout graph. (a) A layout graph. (b)(c) Two template assignment results.

(2) if in x_L^0, S_L needs two conflicts, then $f_0(x_L^0) = |C| + TCost = 2$. Thus $f_2^+(x_L) \le f_0(x_L^0)$.

Hence it holds that $f_2^+(x_L) \le f_0(x_L^0)$ for S_L. For the "T" shape, "Z" shape and "+" shape CIS, we can prove similarly that the statement still holds.

Therefore, for an optimal solution x^0 of P_0, and x obtained from x^0 by omitting the template assignment, it holds that $f_2^+(x) \le f_0(x^0)$. Thus P_2^+ is still a discrete relaxation of problem P_0. □

Similarly, Corollary 1 still holds for problem P_2^+.

We use the Branch and Bound approach in the software package CPLEX [19] to solve problem P_2^+. Since problem P_2^+ is hard to solve in the large scale case, we introduce some graph reduction techniques to reduce the size of the problem, such that it is easy to solve using the Branch and Bound approach. In this paper, the used graph reduction techniques include: connected components calculation, vertices with degree less than 3 removal and 2-edge connected components calculation [20–22]. The vertices with degree less than 3 removal technique will remove some contacts. Since these removed contacts can be easily assigned to templates without any cost of assignment, they would be handled after the template assignment stage in Section 4. Furthermore, for the 2-edge connected components calculation, if vertices i and j are connected by a bridge and are assigned to the same mask at the mask assignment stage, then a conflict is generated between i and j. We can eliminate the conflict by rotating the colors of one of the 2-edge connected components such that i and j are in different masks. Moreover, we do not need to consider template assignment for i and j.

4. Template assignment

After obtaining a discrete relaxation solution of problem P_0 by solving problem P_2^+, we must decide the template assignment for the discrete relaxation solution. The solution of problem P_2^+ divides the initial layout (except removed contacts) into three masks, and we obtain three decomposed layouts L_1, L_2, L_3. Then we should consider the template assignment for every decomposed layout L_m ($m = 1, 2, 3$), which is described as follows:

Template assignment for each decomposed mask

Given: A decomposed contact layout, a set of vertical and horizontal templates.

Find: A template assignment of contacts, which groups some of the contacts by available multi-hole templates.

Subject to: Every contact is assigned to only one of the templates.

Objective: $|C| + \beta \cdot TCost$ is minimized, where $|C|$ is the number of conflicts, and $TCost$ is the total cost of used templates.

For every decomposed layout L_m ($m = 1, 2, 3$), we generate a layout graph LG_m, and consider the template assignment problem on LG_m. The definition of LG_m is:

Definition 6 (*Layout graph*). The layout graph of a layout L_m is a graph $LG_m(C_m, E_{cm})$, where C_m is the set of contacts in the decomposed layout L_m, E_{cm} is the set of conflict edges of the conflict grouping graph $CGG(V, E_c)$ which are vertical or horizontal and connect only contacts in L_m.

Let E_{gm} be the set of grouping edges between any two contacts which are in L_m. Obviously, $E_{gm} \subset E_{cm}$. First we compute all connected components CC of layout graph LG_m ($m = 1, 2, 3$), and then deal with every connected component one by one. Since the edges in E_{cm} are either vertical or horizontal, the considered contacts in every connected component of $LG_m(C_m, E_{cm})$ are lined up vertically or horizontally. Fig. 9(a) shows a connected component with 12 contacts of a layout graph LG_m, where the dotted lines are grouping edges E_{gm}.

There exists some corner contacts in a CC. A corner contact may be one of the three types: i) it is a corner contact belonging to a "+" shape structure; ii) it is a corner contact belonging to a "T" shape structure; iii) it is a corner contact belonging to an "L" shape structure. In Fig. 9(a), i_5 is a corner contact belonging to a "+" shape structure, i_7 is a corner contact belonging to a "T" shape structure, and i_{12} is a corner contact belonging to an "L" shape structure.

For every isolated vertex in LG_m, it is assigned to a single-hole template. For every CC without corner contact, all contacts in CC are lined up vertically or horizontally, and we assign these contacts greedily to a template with the most holes first. Otherwise, for the other complicated CC of LG_m, we consider a heuristic assigning method as follows.

Note that, once all corner contacts in a CC have been assigned to vertical or horizontal templates, then the other contacts can be assigned optimally using the method for a CC without corner contact. Hence, to obtain an optimal template assignment for a complicated CC, we only need to decide whether a corner contact i is assigned to a vertical or a horizontal template. Binary variable y_i is used to indicate if i is assigned to a vertical template or not. That is, $y_i = 0$ denotes that i is assigned to a vertical template; $y_i = 1$ denotes that i is assigned to a horizontal template.

In the experiments, the size of every CC is very small. Hence we check all possible solutions of y to find an optimal template assignment. It must be noted that, if the size of CC is large, then we may find a good solution by some greedy tricks or local search algorithms.

Figs. 9(b) and (c) show two template assignment results of Fig. 9(a). When y is $(0, 1, 0)$, then the result is shown as Fig. 9(b), and $|C| = 5$, $\beta \cdot TCost = 0.01 \times (cost_{T_4} + 3cost_{T_2} + 2cost_{T_1})$. When y is $(0, 0, 1)$, then the result is shown as Fig. 9(c), and $|C| = 4$, $\beta \cdot TCost = 0.01 \times (3cost_{T_3} + cost_{T_2} + cost_{T_1})$.

5. Experimental results

Our discrete relaxation based mask and template assignment method for DSA with TPL of general layout is programmed in C++, and run on a personal computer with 2.4 GHz CPU, 4 GB memory and the Linux operating system. We test our method on the benchmarks provided by Kuang et al. [13]. The width of contacts and the minimum conflict spacing are scaled to 10 nm to reflect the pitch in advanced technology nodes.

Table 1
Statistics of benchmarks for mask and template assignment for DSA with TPL.

Design	$\|V\|$	First experiment			Second experiment		
		$\|E_c\|$	Ratio	$\|E_g\|$	$\|E_c\|$	Ratio	$\|E_g\|$
dp1_Via1	307,739	203,073	0.66	53,120	373,415	1.21	53,228
dp1_Via2	256,885	174,502	0.68	33,473	333,247	1.30	33,473
ed1_Via1	400,123	186,480	0.47	56,450	370,029	0.92	56,450
ed1_Via2	301,607	119,797	0.40	24,587	228,241	0.76	24,587
fft_Via1	99,509	61,926	0.62	16,306	113,993	1.15	16,308
fft_Via2	90,114	62,944	0.70	12,456	117,854	1.31	12,456
mm_Via1	429,664	267,546	0.62	65,426	487,013	1.13	65,585
mm_Via2	341,789	218,668	0.64	39,882	409,226	1.20	39,887
pb1_Via1	79,635	44,668	0.56	11,684	82,017	1.03	11,719
pb1_Via2	59,110	30,518	0.52	6752	58,036	0.98	6752
Avg.	236,617	137,012	0.59	32,013	257,307	1.10	32,044
Ratio		1.00	1.00	1.00	1.87	1.87	1.001

In order to evaluate our method, we design two experiments with different minimum conflict spacings. In the first experiment, the minimum conflict spacing d_c is set as 41 nm, the minimum grouping spacing $d_{g_{min}}$ and the maximum grouping spacing $d_{g_{max}}$ are set as 10 nm and 30 nm, respectively, as in [13]. Note that, increasing d_c has the same effect as shrinking the sizes of nodes. In the second experiment, the minimum conflict spacing d_c is set as 51 nm, the minimum grouping spacing $d_{g_{min}}$ and the maximum grouping spacing $d_{g_{max}}$ are set as 10 nm and 40 nm, respectively. For simplification, the costs of templates are set as $cost_{T_1} = 0$, $cost_{T_2} = 3$, $cost_{T_k} = 2k - 1$, $k = 3, 4, \ldots, K$, in our experiments.

Statistics of the two experiments are listed in Table 1. In the table, for each benchmark, every data in the column $|V|$ is the number of contacts, and every data in the columns $|E_c|$ or $|E_g|$ is the number of conflict edges or grouping edges in the conflict grouping graph CGG, respectively. Moreover, every data in the column "Ratio" is the ratio of the number of conflict edges to the number of contacts for every benchmark. From the row "Ratio", it can be seen that the number of conflict edges in the second experiment is almost 1.87 × the number in the first experiment, while the numbers of grouping edges of the two experiments are almost the same.

5.1. First experiment

In [13], the listed experimental results only use T_2 template to guide contacts for experimental comparisons. Hence, in this experiment, we also only use T_2 template for fair comparisons. In the mask assignment stage, we delete all negative edges he_{ij} in the set E_n. And in the template assignment stage, regardless of the techniques in Section 4, given a layout graph, we generate guiding templates T_2 from left to right and up to down of the positions of the contacts. We group as many as possible the contacts by T_2 templates, and then the rest contacts are guided by single-hole templates. This trick is a greedy approach which would find an optimal assignment. Fig. 10 shows a T_2 template only assignment for the layout graph in Fig. 9(a).

We run the binary files of DAC'15 [12], ILP [12] and ASP-DAC'16 [13] provided by Dr. Kuang. The comparison results of DAC'15 [12], ILP [12], ASP-DAC'16 [13] and ours are listed in Table 2. In Table 2, every data in the column "$|C|$" is the number of conflicts, and every data in the column "$|T_2|$" is the number of used T_2 templates for every benchmark. Moreover, the data in the columns "COST" are the total cost of decomposition for DSA with TPL, which are calculated by COST = $|C| + 0.01 \times cost_{T_2} \times |T_2|$. From the four columns "COST", we can see that, comparing with DAC'15, the results of ILP, ASP-DAC'16 and Ours are significantly better for the mask and template assignment problem of DSA-TP.

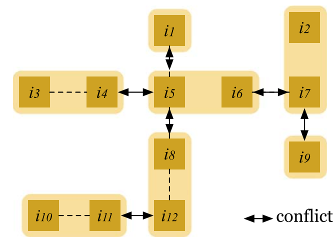

Fig. 10. T_2 template only assignment for the layout graph in Fig. 9(a).

For all benchmarks with $d_c = 41\,nm$ and $d_{g_{max}} = 30\,nm$, ILP, ASP-DAC'16 and our method get optimal solutions. In fact, every data in the column "DRS" is the optimal value of problem P_2^+ for every benchmark, which is a lower bound on the optimal value of problem P_0, since P_2^+ is a discrete relaxation of problem P_0. By the similar claim as Corollary 1, if the assignment cost (column "COST") is equal to the objective value of P_2^+ (column "DRS"), then we obtain an optimal mask and template assignment. Comparing columns "COST" in "ILP", "ASP-DAC'16" and "Ours" with column "DRS", it can be seen that most of the results of the four columns are equal, which means that all the three methods obtain optimal solutions for most of the benchmarks with $d_c = 41\,nm$ and $d_{g_{max}} = 30\,nm$.

Although the method in ILP [12] can evaluate the quality of the experimental results as our method, it can be found that the computing time of the ILP is 73 × more than ours. In addition, although the method in ASP-DAC'16 [13] is about twice faster than our method, it cannot evaluate the quality of the results. This is due to that, the simultaneous assignment method for DSA with TPL in [13] achieves decomposition basing on an off-line Look-Up Table for matching 3-colorable sub-graphs. When the number of nodes is small, the method can quickly match all possible 3-colorable sub-graphs. However, if the number of nodes is large, the Look-Up Table would be too large to match all sub-graphs and cannot find all possible 3-colorable sub-graphs. In order to show the issue of the simultaneous assignment method [13], we design another experiment in the following subsection.

5.2. Second experiment

To further evaluate the effectiveness of our assignment method on more dense layout, we perform another experiment on the benchmarks with $d_c = 51\,nm$ and $d_{g_{max}} = 40\,nm$. The comparison results are listed in Table 3. In the table, $|T_3|$ and $|T_4|$ are the numbers of used T_3 templates and T_4 templates, respectively, and T_Cost is the total cost of used templates, which is calculated by T_Cost = $0.01 \times (cost_{T_2} \times |T_2| + cost_{T_3} \times |T_3| + cost_{T_4} \times |T_4|)$. The notation COST is the total cost by the respective method for the assignment problem, and is calculated by COST = $|C|$+T_Cost. The other notations are the same as those in the first experiment.

The binary files of [13] only use T_2 and T_3 templates to guide the group contacts, while our method does not restrict which type of template is used. Since the experimental results of our method only contain templates T_k, $k = 2, 3, 4$, for fair comparisons, every T_4 template is further split into a T_3 template and a single-hole template, like Fig. 3(b). It is obvious that a conflict would be generated by this split. Correspondingly, for our method the cost of used T_4 templates, i.e., $0.01 \times cost_{T_4} \times |T_4|$, is replaced by the cost of used T_3 templates and the cost of generated conflicts, i.e., $(0.01 \times cost_{T_3} + 1)|T_4|$. Then COST $_1$ of our method is calculated by COST $_1$ = $|C| + T\,Cost - 0.01 \times cost_{T_4} \times |T_4| + (0.01 \times cost_{T_3} + 1)|T_4| = |C| + T\,Cost + 0.98|T_4|$.

First, we compare the column "COST" in "ASP-DAC'16" with the column "COST$_1$" in "Ours". Assuming that only T_2 and T_3 templates are used, we can reduce the total cost of decomposition by 6%, and for every benchmark, we achieve a better result. Moreover, we can reduce

Table 2
Comparison results of mask and template assignment for DSA with TPL, $d_c = 41\ nm$, $d_{g_{min}} = 10\ nm$, $d_{g_{max}} = 30\ nm$.

Design	DAC'15 [12]				ILP [12]				ASP-DAC'16 [13]				Ours				Lower
	$\|C\|$	$\|T_2\|$	COST	CPU(s)	$\|C\|$	$\|T_2\|$	COST	CPU(s)	$\|C\|$	$\|T_2\|$	COST	CPU(s)	$\|C\|$	$\|T_2\|$	COST	CPU(s)	DRS
dp1_Via1	13	29,757	905.71	1.80	10	24	10.72	164.99	10	24	10.72	1.64	10	24	10.72	3.74	10.72
dp1_Via2	24	18,191	569.73	1.59	0	400	12	498.37	0	400	12	2.91	0	400	12	3.30	12
ed1_Via1	0	34,233	1026.99	2.17	0	1	0.03	215.98	0	1	0.03	1.77	0	1	0.03	5.26	0.03
ed1_Via2	3	16,395	494.85	1.60	0	90	2.7	168.42	0	90	2.7	2.61	0	90	2.7	3.92	2.7
fft_Via1	0	9183	275.49	0.76	0	0	0	53.65	0	0	0	0.59	0	0	0	1.28	0
fft_Via2	11	6621	209.63	0.70	0	175	5.25	515.29	0	175	5.25	1.43	0	175	5.25	1.15	5.25
mm_Via1	14	37,897	1150.91	2.55	11	25	11.75	220.92	11	25	11.75	1.94	11	25	11.75	5.56	11.75
mm_Via2	18	22,361	688.83	2.01	0	384	11.52	297.92	0	384	11.52	3.00	0	384	11.52	4.43	11.52
pb1_Via1	1	7191	216.73	0.63	1	9	1.27	38.8	1	9	1.27	0.49	1	9	1.27	1.03	1.27
pb1_Via2	4	40,044	1205.32	0.53	0	49	1.47	49.3	0	49	1.47	1.03	0	49	1.47	0.76	1.44
Avg.	9	22,187	674.42	1.43	2	115	5.67	222.36	2	115	5.67	1.74	2	115	5.67	3.05	5.67
Ratio	4.19	191.60	120.99	0.47	1.00	1.00	1.00	73.02	1.00	1.00	1.00	0.57	1.00	1.00	1.00	1.00	1.00

Table 3
Comparison results of mask and template assignment for DSA with TPL, $d_c = 51\ nm$, $d_{g_{min}} = 10\ nm$, $d_{g_{max}} = 40\ nm$.

Design	ASP-DAC'16 [13]						Ours								Lower
	$\|C\|$	$\|T_2\|$	$\|T_3\|$	T_Cost	COST	CPU(s)	$\|C\|$	$\|T_2\|$	$\|T_3\|$	$\|T_4\|$	T_Cost	COST_1	COST	CPU(s)	DRS
dp1_Via1	3690	5219	42	158.67	3848.67	67.5	3413	5100	137	6	160.27	3579.15	3573.27	31.62	3557.61
dp1_Via2	7422	6189	74	189.37	7611.37	378.5	6943	6128	178	11	193.51	7147.29	7136.51	61.36	7108.42
ed1_Via1	2297	3395	18	102.75	2399.75	40.46	2159	3479	52	3	107.18	2269.12	2266.18	29.01	2258.25
ed1_Via2	1667	1998	6	60.24	1727.24	39.74	1572	1993	24	0	60.99	1632.99	1632.99	22.60	1631.69
fft_Via1	743	1277	8	38.71	781.71	11.18	694	1281	30	1	40.00	734.98	734.00	8.57	730.86
fft_Via2	2544	2453	40	75.59	2619.59	110.3	2380	2429	67	2	76.36	2458.32	2456.36	28.43	2447.43
mm_Via1	3729	5464	22	165.02	3894.02	61.36	3576	5398	94	2	166.78	3744.74	3742.78	37.18	3732.44
mm_Via2	8165	6745	84	206.55	8371.55	381.3	7638	6678	202	13	211.35	7862.09	7849.35	86.10	7811.86
pb1_Via1	752	866	2	26.08	778.08	10.53	727	851	11	0	26.08	753.08	753.08	5.72	751.58
pb1_Via2	833	729	8	22.27	855.27	46.77	737	699	10	0	21.47	758.47	758.47	7.39	755.97
Avg.	3184	3433	30	104.5	3288.7	114.7	2984	3403	80	4	106.4	3094.0	3090.3	31.8	3078.6
Ratio	1.07	1.01	0.38	0.98	1.06	3.61	1.00	1.00	1.00		1.00	1.00	1.004	1.00	1.00

Fig. 11. Mask and template assignment for benchmark dp1_Via1 with $d_c = 51\ nm$ and $d_{g_{max}} = 40\ nm$.

the total number of conflicts by 7%. In addition, the CPU time of ASP-DAC'16 is 3.61 × more than ours. Then, we compare the column “COST” in “Ours” with the column “DRS”, our total cost averagely is only 0.4% greater than the lower bound. Hence, the gap between our total cost and the optimal value should be less than 0.4%. The experimental results indicate that our method almost obtains optimal solutions for all benchmarks. At last, a part of the experimental result for the benchmark dp1_Via1 with d_c = 51 nm and d_{gmax} = 40 nm is shown as Fig. 11.

It must be noted that the quality evaluation of a solution is significant for an NP-hard problem which has real applications. If we know the gap between the obtained result and the optimal value, then we know whether an instance of the problem is solved or not. Typically, for this experiment with denser setting, our method shows that the gap between our result and the optimal value is very tiny, but the number of conflicts is still large. This indicates that one or more masks might be needed for further eliminating the conflicts.

6. Conclusions

In this paper, we consider the contact layer mask and template assignment problem of DSA with TPL for general layout, and propose a discrete relaxation method. First, we introduce negative edges in the conflict grouping graph, and weight the edges of the conflict grouping graph. Then we formulate a discrete relaxation problem of the contact layer assignment problem of DSA with TPL. For obtaining better results, we introduce triangle edges in the weighted conflict grouping graph, and thus introduce some valid inequalities in the discrete relaxation problem. We transform the discrete relaxation solution to a legal solution of the initial problem by addressing the template assignment problem on the layout graph. Our discrete relaxation based method can estimate the gap between the obtained solution and the optimal solution in the experiment, which is meaningful for the NP-hard problem. Furthermore, our experimental results show that the gaps between the obtained solutions and the optimal solutions are very small. Specially, the discrete relaxation approach verifies the optimality of our experimental results of sparse benchmarks since the gaps are 0. Finally, it must be remarked that we only consider the 1-D templates in this paper. However, the proposed method can be extended to handle more general templates like 2 × 2, which needs further careful investigation.

Acknowledgments

The authors would like to thank Dr. Jian Kuang for the discussions and providing the binaries and the benchmarks. The authors also would like to thank the anonymous reviewers, whose insightful comments and suggestions are rather helpful for improving the quality of the manuscript.

References

[1] D.Z. Pan, B. Yu, J.-R. Gao, Design for manufacturing with emerging nanolithography, IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 32 (10) (2013) 1453–1472.
[2] Y.W. Chang, R.G. Liu, S.Y. Fang, Euv and e-beam manufacturability: challenges and solutions, in: Proceedings of the ACM/IEEE Design Automation Conference (DAC), San Francisco, CA, 2015.
[3] C.M. Bates, T. Seshimo, M.J. Maher, W.J. Durand, J.D. Cushen, L.M. Dean, G. Blachut, C.J. Ellison, C.G. Willson, Polarity-switching top coats enable orientation of sub-10-nm block copolymer domains, Science 338 (6108) (2012) 775–779.
[4] X.Y. Bao, H. Yi, C. Bencher, L.W. Chang, H. Dai, Y. Chen, P.T. Chen, H.S. Wong, Sram, nand, dram contact hole patterning using block copolymer directed self-assembly guided by small topographical templates, in: Proceedings of the 2011 IEEE International Electron Devices Meeting (IEDM), 2011.
[5] H. Yi, X.Y. Bao, J. Zhang, C. Bencher, L.W. Chang, X. Chen, R. Tiberio, J. Conway, H. Dai, Y. Chen, S. Mitra, H.-S.P. Wong, Flexible control of block copolymer directed self assembly using small, topographical templates: potential lithography solution for integrated circuit contact hole patterning, Adv. Mater. 24 (23) (2012) 3107–3114.
[6] B. Yu, X. Xu, S. Roy, Y. Lin, J. Ou, D.Z. Pan, Design for manufacturability and reliability in extreme-scaling vlsi, Sci. China Inf. Sci. 59 (061406) (2016) 1–23.
[7] R. Ruiz, H. Kang, F.A. Detcheverry, E. Dobisz, D.S. Kercher, T.R. Albrecht, J.J. de Pablo, P.F. Nealey, Density multiplication and improved lithography by directed block copolymer assembly, Science 321 (5891) (2008) 936–939.
[8] Z. Xiao, Y. Du, M.D.F. Wong, Y. He, H.S.P. Wong, H. Zhang, Contact pitch and location prediction for directed self-assembly template verification, in: Proceedings of the IEEE/ACM Asia and South Pacific Design Automation Conference (ASP-DAC), Chiba/Tokyo, 2015, pp. 644–651.
[9] R.A. Griffiths, A. Williams, C. Oakland, J. Roberts, A. Vijayaraghavan, T. Thomson, Directed self-assembly of block copolymers for use in bit patterned media fabrication, J. Phys. D: Appl. Phys. 46 (50) (2013) 503001.
[10] S. Shim, W. Chung, Y. Shin, Redundant via insertion for multiple-patterning directed-self-assembly lithography, in: Proceedings of the ACM/IEEE Design Automation Conference (DAC), no. 41, San Francisco, CA, 2016.
[11] Z. Xiao, C.X. Lin, M.D.F. Wong, H. Zhang, Contact layer decomposition to enable dsa with multi-patterning technique for standard cell based layout, in: Proceedings of the IEEE/ACM and South Pacific Design Automation Conference (ASP-DAC), vol. 95–102, Macao, 2016.
[12] Y. Badr, A. Torres, P. Gupta, Mask assignment and synthesis of dsa-mp hybrid lithography for sub-7nm contacts/vias, in: Proceedings of the ACM/IEEE Design Automation Conference (DAC), San Francisco, CA, 2015, pp. 1–6.
[13] J. Kuang, J.J. Ye, F.Y. Young, Simultaneous template optimization and mask assignment for dsa with multiple patterning, in: Proceedings of the IEEE/ACM Asia and South Pacific Design Automation Conference (ASP-DAC), 2016, pp. 75–82.
[14] X. Li, Z. Zhu, W. Zhu, Discrete relaxation method for triple patterning lithography layout decomposition, IEEE Trans. Comput. 66 (2) (2017) 258–298.
[15] Y. Du, D. Guo, M.D. Wong, H. Yi, H.-S. P. Wong, H. Zhang, Q. Ma, Block copolymer directed self-assembly (dsa) aware contact layer optimization for 10 nm 1d standard cell library, in: Proceedings of IEEE/ACM International Conference on Computer-Aided Design (ICCAD), San Jose, CA, 2013, pp. 186–193.
[16] Z. Xiao, D. Guy, M.D. Wong, H. Yi, M.C. Tung, H.S.P. Wong, Layout optimization and template pattern verification for directed self-assembly (dsa), in: Proceedings of the ACM/IEEE Design Automation Conference (DAC), no. 199, San Francisco, CA, 2015.
[17] S.-Y. Fang, Y.-X. Hong, Y.-Z. Lu, Simultaneous guiding template optimization and redundant via insertion for directed self-assembly, IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 36 (1) (2017) 156–169.
[18] J. Ou, B. Yu, D.Z. Pan, Concurrent guiding template assignment and redundant via insertion for dsa-mp hybrid lithography, in: Proceedings of the ACM International Symposium on Physical Design (ISPD), Sonoma, CA, 2016, pp. 39–46.
[19] CPLEX, (2016, July). [Online]. Available: ⟨http://www-01.ibm.com/⟩.
[20] B. Yu, K. Yuan, B. Zhang, D. Ding, D.Z. Pan, Layout decomposition for triple patterning lithography, IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 34 (3) (2015) 433–446.
[21] S.Y. Fang, Y.W. Chang, W.Y. Chen, A novel layout decomposition algorithm for triple patterning lithography, IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 33 (3) (2014) 397–408.
[22] J. Kuang, E.F. Young, An efficient layout decomposition approach for triple patterning lithography, in: Proceedings of the ACM/IEEE Design Automation Conference (DAC), no. 69, San Francisco, CA, 2013.