Analytical Mixed-Cell-Height Legalization Considering Average and Maximum Movement Minimization*

Xingquan Li
School of Mathematics and Statistics,
Minnan Normal University
Zhangzhou, 363000, China
xqli@mnnu.edu.cn

Jianli Chen
Center for Discrete Mathematics and Theoretical Computer Science, Fuzhou University
Fuzhou, 350108, China
jlchen@fzu.edu.cn

Wenxing Zhu
Center for Discrete Mathematics and Theoretical Computer Science, Fuzhou University
Fuzhou, China
wxzhu@fzu.edu.cn

Yao-Wen Chang
Graduate Institute of Electronics Engineering,
Department of Electrical Engineering,
National Taiwan University
Taipei, 106, Taiwan
ywchang@ntu.edu.tw

ABSTRACT

Modern circuit designs often contain standard cells of different row heights to meet various design requirements. Due to the higher interference among heterogeneous cell structures, the legalization problem for mixed-cell-height standard cells becomes more challenging. In this paper, we present an analytical legalization algorithm for mixed-cell-height standard cells to simultaneously minimize the average and the maximum cell movements. We formulate it as a mixed integer quadratic programming problem (MIQP), which allows cell spreading concurrently in both the horizontal and vertical directions. By relaxing its discrete constraints to linear ones, we convert the MIQP into a quadratic programming problem (QP). To solve the QP efficiently, we further reformulate it as a linear complementarity problem (LCP), and solve the LCP by a modulus-based matrix splitting iteration method (MMSIM). To guarantee the convergence of the MMSIM and the equivalence between the QP and the LCP, we use a series of operations to ensure that its induced objective matrix is symmetric positive definite and its constraint matrix is of full row rank. Experimental results demonstrate the effectiveness of our algorithm in reducing both the average and the maximum cell movements for mixed-cell-height legalization.

CCS CONCEPTS

• **Computer systems organization → VLSI circuit design automation**; Physical design; • **Placement** → Legalization;

KEYWORDS

Mixed-cell-height, legalization, VLSI placement, quadratic programming, linear complementarity problem

*This work was supported by Empyrean and the National Natural Science Foundation of China under Grants 11501115, 61672005 and 11331003, and by AnaGlobe, TSMC, MOST of Taiwan under Grant No's MOST 105-2221-E-002-190-MY3, MOST 106-2911-I-002-511, and MOST 107-2221-E-002-161-MY3.

ISPD '19, April 14–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6253-5/19/04…$15.00
https://doi.org/10.1145/3299902.3309750

ACM Reference Format:
Xingquan Li, Jianli Chen, Wenxing Zhu, and Yao-Wen Chang. 2019. Analytical Mixed-Cell-Height Legalization Considering Average and Maximum Movement Minimization. In *2019 International Symposium on Physical Design (ISPD '19), April 14–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3299902.3309750

1 INTRODUCTION

Modern circuit designs often contain (tens of) millions of standard cells located at placement sites on rows. To meet various design requirements such as low power and high performance, multi-deck cells occupying multi-rows (e.g., flip-flops) are often used in advanced technologies [2, 11]. Such multi-row height standard cells bring up challenging issues for placement, especially the mixed-cell-height legalization, due to their heterogenous cell structures and additional power-rail constraints, as pointed out in [7, 17].

In traditional single-row height standard-cell legalization, cell overlapping is independent among rows. In contrast, with multi-row height cells, shifting a cell in one row may cause cell overlaps in another row. The heterogenous cell structures could incur substantial global cell interferences among all cells in a circuit. Due to the global cell interference, existing single-row height standard-cell legalizers [4, 6, 8, 10, 13] cannot directly be extended to handle mixed-cell-height standard cells effectively. As a result, a mixed-cell-height legalization method needs to consider the heterogenous cell structures, with more global cell interferences and larger solution spaces. Moreover, the alignment of power (VDD) or ground (VSS) lines must be considered in mixed-cell-height standard-cell legalization. For single-row height standard-cell legalization, such VDD/VSS alignment can easily be handled by vertical cell flipping, for example, the single-row height cell c_1 in Figure 1. However, the vertical cell flipping is invalid for an even-row height cell (e.g., the double-row cell c_2 in Figure 1). During legalization, therefore, each even-row height cell must be aligned to its correct row which meets the VDD/VSS constraint.

In addition, to preserve the quality of a given global placement, an ideal legalization method should minimize not only the average cell movement but also the maximum one [9]; see Figure 2 for an illustration. In Figure 2(a), if we focus only on minimizing the average movement, we may obtain a legalization result as in Figure 2(b); in contrast, if we minimize the average and maximum cell movements simultaneously, we can obtain a better legalization result as

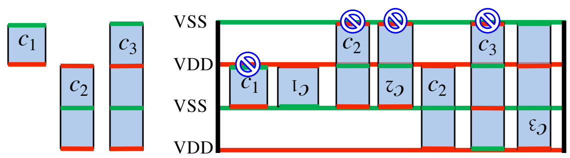


Figure 1: Example of the VDD/VSS alignment constraints.

in Figure 2(c). Both of the results in Figures 2(b) and 2(c) have the same average cell movement, but the maximum cell movement of the result in Figure 2(b) is twice of that in Figure 2(c). In this paper, we aim to minimize the average and maximum cell movements simultaneously for mixed-cell-height legalization.

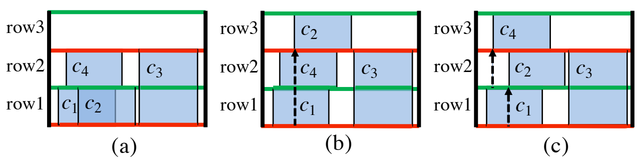


Figure 2: Comparisons on legalization with average cell movement and that with simultaneous average and maximum cell movement.

Recent state-of-the-art works considered the mixed-cell-height standard-cell legalization problem [5, 7, 12, 16, 17]. Wu et al. [17] first investigated the standard-cell legalization with both of single- and double-row height cells. In [7], a multi-row local legalization algorithm was proposed to place cells in a local region. Wang et al. [16] extended Abacus to handle the legalization problem with mixed-cell-height standard cells. Hung et al. [12] proposed a flow-based method to spread cells and placed cells based on an integer linear program (ILP). With a modulus-based matrix splitting iteration method (MMSIM), Chen et al. [5] developed a near optimal legalization method to address mixed-cell-height standard-cell legalization. To guarantee the MMSIM convergence, the authors pointed out that the objective matrix should be symmetric positive definite, and the constraint matrix should be of full row rank. Nevertheless, these legalization methods focus only on minimizing the average cell movement, and do not consider the maximum cell movement.

To minimize the maximum cell movement for single-row height standard-cell legalization, Darav et al. [8] proposed a flow-based legalization method by finding augmentation paths among bins. If the flow of a candidate augmentation path is larger than a preset value (named *maximum cell movement*), it would be pruned. Apparently, their method considers the maximum cell movement as a hard constraint rather than an objective. Further, in order to resolve cell overlapping, cells are moved from a dense bin to a sparse one along paths [8]. In single-row height standard-cell legalization, cell overlapping is independent among rows. With multi-row height cells, in contrast, shifting a cell in one row may cause cell overlaps in another row. What is worse, shifting a multi-row height cell in a bin to anther one may make cells illegal due to a complex domino effect. Furthermore, in order to meet the VDD/VSS alignment constraints, it is much harder for the flow-based method to control cells movement. As a result, it is not easy to extend the flow-based method in [8] to handle the mixed-cell-height cell legalization problem.

In this paper, we present an analytical mixed-cell-height standard-cell legalization algorithm to simultaneously minimize the average and the maximum cell movements. The major contributions of our work are summarized as follows:

- By analyzing and remodeling the objective function and constraints, we formulate the mixed-cell-height standard-cell legalization problem as a mixed integer quadratic program (MIQP), which considers not only the average cell movement, but also the maximum cell movement, the sub-maximum movement, and the third maximum movement, etc.
- By relaxing its discrete constraints to linear ones, we convert the MIQP into a quadratic programming problem (QP). Unlike the work in [5] which minimizes only the average cell movement in the horizontal direction, we consider cells spreading continuously in both the horizontal and vertical directions.
- To solve the QP efficiently, we further reformulate it as a linear complementarity problem (LCP), and solve the LCP by a modulus-based matrix splitting iteration method (MMSIM). We apply a series of operations to guarantee the convergence of the MMSIM and prove the equivalence between the QP and the LCP.
- We propose a linear programming (LP) based method to further minimize the maximum cell movement in the horizontal direction.
- Experimental results show that our legalization model and algorithm are effective for minimizing both the average and the maximum cell movements. Compared with the state-of-the-art work [5] based on the same set of benchmarks used in [5], for example, our algorithm reduces the average and maximum cell movements by 16% and 64%, respectively.

The rest of this paper is organized as follows. Section 2 gives the problem statement. Section 3 shows our legalization model, which allows cell movements in both the horizontal and vertical directions simultaneously. Section 4 details our legalization algorithms. Experimental results are given in Section 5, and finally conclusions are made in Section 6.

2 PROBLEM STATEMENT

The problem given are a global placement result with n' mixed-cell-height standard cells $C = \{c_1, c_2 \cdots, c_{n'}\}$ with the initial bottom-left coordinate (x_i^0, y_i^0) for each cell, and the height and width of a cell c_i denoted as h_i and w_i, respectively. Each cell has a boundary power-rail type, VSS or VDD. The chip is a rectangular sheet from $(0, 0)$ to (W, H), where W and H are the chip width and height, and R_h and S_w denote the row height and placement site width, respectively. In this paper, the mixed-cell-height standard-cell legalization problem aims at placing each cell to its best position, such that the average and maximum cell movements are minimized and the following constraints are satisfied:

(1) cells should be aligned with correct VDD/VSS rails;
(2) cells should be non-overlapping;
(3) cells should be inside the chip;
(4) cells should be located at placement sites on rows.

Let (x_i, y_i) be the bottom-left coordinate of cell c_i, $i = 1, 2, \cdots, n'$. The problem of mixed-cell-height standard-cell legalization with simultaneous average and maximum movements minimization can be formulated as:

$$\min_{x,y} \frac{1}{n'} \sum_{c_i \in C} (|x_i - x_i^0| + |y_i - y_i^0|) + \omega \cdot \max_{c_i \in C}(|x_i - x_i^0| + |y_i - y_i^0|) \quad (1)$$

s.t. $y_i = k_i' R_h, \ \forall c_i \in C,$

$$k_i' \in \begin{cases} \{0, 1, 2, \ldots\} \text{ if } c_i \text{ is of an odd-row height;} \\ \{0, 2, 4, \ldots\} \text{ (VDD boundaries) or} \\ \{1, 3, 5, \ldots\} \text{ (VSS), otherwise;} \end{cases} \quad (1a)$$

$x_i + w_i \leq x_j, \ \forall c_i, c_j \in C,$

if c_i and c_j are in the same row, and $x_i \leq x_j$; (1b)

$0 \leq x_i, x_i + w_i \leq W, \ 0 \leq y_i, y_i + h_i \leq H, \ \forall c_i \in C;$ (1c)

$x_i = l_i S_w, l_i \in \{0, 1, 2, \ldots\}, \ \forall c_i \in C,$ (1d)

where ω is a user-defined weighting parameter for the average and the maximum cell movements.

3 PROBLEM REFORMULATIONS

In this section, we first formulate the mixed-cell-height legalization problem as a mixed integer quadratic programming problem (MIQP), then we convert the MIQP to a quadratic programming problem (QP), and further we reformulate the QP as a linear complementarity problem (LCP).

3.1 Mixed Integer Quadratic Programming (MIQP)

The objective function of Problem (1) is the weighted sum of the average cell movement and the maximum cell movement. We transform the objective as follows:

$$\min_{x,y} \sum_{c_i \in C} \frac{\alpha_i}{2} ((x_i - x_i^0)^2 + (y_i - y_i^0)^2), \quad (2)$$

where α_i can be seen as a weight on the movement of cell c_i. In fact, Objective (2) includes minimizing the maximum cell movement if α_i is assigned a proper value, $i = 1, 2, \cdots, n'$.

Naturally, it would be better that a legalizer can not only reduce the maximum cell movement, but also the sub-maximum movement, the third maximum movement, etc. For example, if a cell c_i overlaps with a macro, in order to resolve the overlap, c_i must be moved out of the macro, then a large cell movement, even the maximum one, is likely generated. If the generated maximum cell movement is much more than the other cell movements, then minimizing the maximum cell movement is meaningless to all the remaining cells. Hence, we handle the above scenario by re-assigning weight α_i to the movement of each cell c_i. An intuitive weight setting rule is that, if the movement of c_i is larger than the movement of c_j, then α_i should be larger than α_j to reduce the movement of c_i. In this paper, α_i is set as follows:

$$\alpha_i = \left(\frac{(x_i - x_i^0)^2 + (y_i - y_i^0)^2}{\frac{1}{n'} \sum_{c_j \in C} (|x_j - x_j^0| + |y_j - y_j^0|)} \right)^{\kappa}, \quad (3)$$

where $\kappa \geq -1$ and is relative to the value of ω in Problem (1). The parameter κ is used to make a trade-off between the average cell movement and the maximum cell movement. If $\kappa \gg 1$, it focuses on minimizing the maximum cell movement; if $\kappa = -1$, it focuses on minimizing the average cell movement. In Equation (3), x_i and y_i are set as the latest iteration results (coordinates) of cell c_i in our algorithm (to be described in Subsection 4).

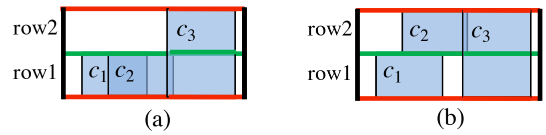


Figure 3: Cells are aligned to the nearest correct rows.

We give the handling of constraints and detailed analysis here. First, for Constraint (1a), cells should be aligned with correct rows to meet the VDD/VSS constraints. In order to obtain a high-quality legalization solution with the least vertical movement, a trivial yet effective operation is moving each cell to the nearest VDD/VSS rail. Ideally, after aligning to the nearest correct row, the y-coordinate of each cell c_i is updated to y_i' from y_i^0. If the distribution of cells in a row is locally sparse, then the overlaps among these cells in this row can be desirably solved; in contrast, if the distribution of cells in a row is locally dense, then the locations of some cells in this row may not be desirable. They should be assigned to other rows for which Constraint (4) still should be satisfied:

$$y_i \in \{y_i' - k_i^l R_h, y_i' + k_i^u R_h\}, \ \forall c_i \in C. \quad (4)$$

In the above equation, k_i^l and k_i^u represent the maximum straight down and straight up movements of a cell c_i, respectively. If c_i is of an odd-row height, then $k_i^l \in \{0, 1, 2, \ldots, \lfloor \frac{y_i'}{R_h} \rfloor\}$ and $k_i^u \in \{0, 1, 2, \ldots, \lfloor \frac{H - y_i'}{R_h} \rfloor\}$; if c_i is of an even-row height, then $k_i^l \in \{0, 2, 4, \ldots, \lfloor \frac{y_i'}{R_h} \rfloor\}$ and $k_i^u \in \{0, 2, 4, \ldots, \lfloor \frac{H - y_i'}{R_h} \rfloor\}$. For example, in Figure 3, after aligning to the nearest correct row, the distribution of cells in Row 1 is too dense to resolve the overlaps. As a result, the cell c_2 should be aligned to Row 2.

Second, for Constraint (1b), all cells in the same row are sorted by their initial bottom-left x^0, i.e., $x_i^0 \leq x_j^0$, if cell c_i is on the left of cell c_j according to the global placement result. Furthermore, if cells c_i and c_j are in different rows, then the position constraint between x_i and x_j is free. Hence, Constraint (1b) can be reformulated as follows:

$$x_i + w_i \leq x_j + M(1 - z_{ij}), \ z_{ij} \in \{0, 1\}, \ \text{if } x_i^0 \leq x_j^0, \quad (5)$$

where M is a large enough number, say, chip width W. If cells c_i and c_j are adjacent in the same row, then $z_{ij} = 1$; otherwise, $z_{ij} = 0$.

Third, for Constraint (1c), for each cell c_i, $0 \leq y_i, y_i + h_i \leq H$, is satisfied under Constraint (1a). Thus, it can be removed. In addition, since we minimize Objective (2), the horizontal moving of each cell would not be far away from its original position, and there would be few cells out of the boundary. We skip the right boundary constraint temporarily, and then Constraint (1c) is changed as follows:

$$x_i \geq 0, \ \forall c_i \in C. \quad (6)$$

Fourth, in Constraint (1d), if all the cells are placed at their best positions in rows, then we only need to shift each cell to its nearest placement site. We will detail Constraint (1d) in Section 4.

Overall, Objective (2) and Constraints (4), (5), and (6) form a mixed integer quadratic programming problem (MIQP).

3.2 Quadratic programming (QP)

For the MIQP in Subsection 3.1, variable y_i in Constraint (4) and variable z_{ij} in Constraint (5) are integral. In addition, in order to obtain a better result, all cells in a circuit should be considered together instead of row-by-row. Consequently, it is hard to solve the MIQP for large-scale circuits. In this subsection, we relax Constraints (4) and (5) to linear constraints. Accordingly, the MIQP is relaxed to a quadratic programming problem.

In the work [3], Bai proposed a modulus-based matrix splitting iteration method (MMSIM), which is very efficient for solving linear complementarity problems. In addition, Chen et al. [5] applied the MMSIM to the QP legalization problem. In order to use the effective and efficient MMSIM, as in [5], we split all multi-row height cells into single-row ones for the MMSIM solver. Then the mixed-cell-height standard cells $C = \{c_1, c_2 \cdots, c_{n'}\}$ are split as single-row sub-cells $SC = \{sc_1, sc_2 \cdots, sc_n\}$. These single-row height sub-cells should satisfy:

$$\begin{aligned} x_{i1} &= x_{i2} = \cdots = x_{ir_i}, \\ y_{i1} + (r_i - 1)R_h &= y_{i2} + (r_i - 2)R_h = \cdots = y_{ir_i}. \end{aligned} \tag{7}$$

where sub-cells $sc_{i1}, sc_{i2}, \ldots, sc_{ir_i}$ are split from an r_i-row height cell, $r_i = h_i/R_h$.

After cell splitting, Objective (2) is transformed to

$$\min_{x,y} \sum_{sc_i \in SC} \frac{\alpha_i}{2r_i}((x_i - x_i^0)^2 + (y_i - y_i^0)^2). \tag{8}$$

Since an r_i-row height cell is split into r_i single-row height sub-cells and the movement of the r_i-row cell is counted r_i times, we divide the objective function by r_i for each cell in Objective (8).

Next, we relax Constraints (4) and (5) to linear constraints. Actually, for each sub-cell sc_i, after aligning it to its nearest correct row, i.e., $y_i = y_i'$, the movement of sub-cell sc_i in the vertical direction is minimized. Since sub-cells may be re-assigned to other rows to achieve a better legalization solution, we relax the range y_i of sub-cell sc_i from an integer to a real number. That is

$$y_i \in [y_i' - k_i^l R_h, y_i' + k_i^u R_h], \ \forall sc_i \in SC, \tag{9}$$

where $k_i^l \in \{0, 1, 2, \ldots, \lfloor \frac{y_i'}{R_h} \rfloor\}$ and $k_i^u \in \{0, 1, 2, \ldots, \lfloor \frac{H - y_i'}{R_h} \rfloor\}$.

Since the moving range $y_i \in [y_i' - k_i^l R_h, y_i' + k_i^u R_h]$ of sub-cell sc_i is excessive, we can speed up the process by restricting the moving orientation o_i to be vertical (upward or downward) for each sub-cell sc_i. Then, Constraint (9) of sc_i is limited as followed:

$$y_i \in \begin{cases} \left[y_i' - k_i^l R_h, y_i'\right], & \text{if } o_i \text{ is downward;} \\ \left[y_i', y_i' + k_i^u R_h\right], & \text{if } o_i \text{ is upward,} \end{cases} \tag{10}$$

where k_i^l and k_i^u are used to control the range of y_i. Correspondingly, the vertical moving interval VMI_i of sub-cell sc_i is $[y_i' - k_i^l R_h, y_i' + R_h]$ or $[y_i', y_i' + k_i^u R_h]$. The setting of sub-cell orientation is detailed in Algorithm 1 (Section 4). For example, as shown in Figure 4(a), if the orientations of sub-cells sc_1, sc_2, sc_3, sc_4 and sc_5 are upward, downward, upward, upward, and upward, respectively, and $k_1^u = k_2^l = k_5^u = 1$ and $k_3^u = k_4^u = 2$, then the vertical moving interval of each sub-cell is marked by a doubly headed arrow line shown in Figure 4(a), and the corresponding range of y_i for sub-cell sc_i is marked by a square bracket in Figure 4(b).

Next, the mixed integer constraint (5) would be relaxed to a linear constraint. According to Constraint (5), if $x_i^0 \le x_j^0$, and sc_i and sc_j are adjacent, then $x_i + w_i \le x_j + M(1 - z_{ij})$, where $z_{ij} \in \{0, 1\}$ denotes whether two adjacent sub-cells sc_i and sc_j are in the same row. Since all the sub-cells have the same height (i.e., $h_i = R_h$), $z_{ij}R_h$ reflects the overlapping length of sub-cells sc_i and sc_j in the

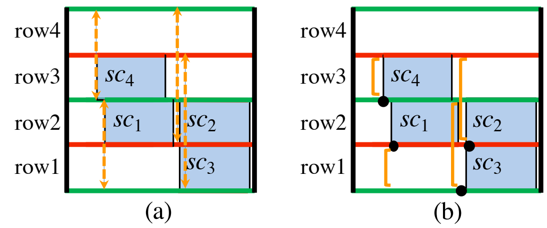


Figure 4: (a) Vertical moving intervals of sub-cells. (b) The moving ranges of bottom-left y-coordinate of sub-cells.

vertical direction. If $z_{ij} = 1$, i.e., $|y_i - y_j| = 0$, then the vertical overlapping length is R_h; if $z_{ij} = 0$, i.e., $|y_i - y_j| \ge R_h$, then the vertical overlapping length is 0. The vertical overlapping length of sub-cells sc_i and sc_j with $z_{ij}R_h = \max\{R_h - |y_i - y_j|, 0\}$. Then, for two adjacent sub-cells sc_i and sc_j, we have

$$z_{ij} = \max\left\{1 - \frac{|y_i - y_j|}{R_h}, 0\right\}, \text{ and} \tag{11}$$

$$x_i + w_i \le x_j + M \cdot \min\left\{\frac{|y_i - y_j|}{R_h}, 1\right\}, \text{ if } x_i^0 \le x_j^0. \tag{12}$$

By Equation (11), the integer variable z_{ij} is relaxed to a continuous variable. In addition, since the VMI_i of each sub-cell sc_i can be pre-calculated, if $VMI_i \cap VMI_j = \emptyset$, then we do not need to consider Constraint (12). For example, for sub-cells sc_2 and sc_5 in Figure 4(a), $VMI_2 \cap VMI_5 = \emptyset$. Otherwise, if $VMI_i \cap VMI_j \ne \emptyset$ for sub-cells sc_i and sc_j, then we would check wether sc_i and sc_j are adjacent.

Observing Constraint (12), it can be seen that this constraint is still hard to handle due to the absolute value function and the minimum one. It should further be transformed to a linear constraint by eliminating the absolute and the minimum value functions.

For two adjacent sub-cells sc_i and sc_j, where $VMI_i \cap VMI_j \ne \emptyset$, there exist four possible cases: (1) sc_i and sc_j are in the same row and have the same orientation, as shown in Figure 5(a); (2) sc_i and sc_j are in the same row and have different orientations, as in Figure 5(b); (3) sc_i and sc_j are in different rows and have the same orientation, as in Figure 5(c); (4) sc_i and sc_j are in different rows and have different orientations, as in Figure 5(d);

For Cases (2) and (3), the relationship between y_i and y_j is known. If $y_i \le y_j$, then $|y_i - y_j| = y_j - y_i$; if $y_i > y_j$, then $|y_i - y_j| = y_i - y_j$. However, for Cases (1) and (4), we do not know which is larger between y_i and y_j. For the two cases, we use the initial y-coordinates to determine the value of $|y_i - y_j|$. That is, if $y_i^0 \le y_j^0$, then $|y_i - y_j| = y_j - y_i$; otherwise, $|y_i - y_j| = y_i - y_j$. We introduce the notation $\ominus$ to denote the operator between y_i and y_j, and let

$$y_i \ominus y_j = \begin{cases} y_i - y_j, & \text{if } y_i \ge y_j \text{ for Cases (2) and (3),} \\ & \text{and if } y_i^0 \ge y_j^0 \text{ for Cases (1) and (4);} \\ y_j - y_i, & \text{if } y_i < y_j \text{ for Cases (2) and (3),} \\ & \text{and if } y_i^0 < y_j^0 \text{ for Cases (1) and (4).} \end{cases} \tag{13}$$

Furthermore, since the maximum value of $\frac{y_i \ominus y_j}{R_h}$ is $k_i + k_j$ (where $k_i = k_i^l$ if o_i is downward; otherwise $k_i = k_i^u$), we have $\min\{\frac{y_i \ominus y_j}{(k_i + k_j)R_h}, 1\} = \frac{y_i \ominus y_j}{(k_i + k_j)R_h}$. After these transformations, for two adjacent sub-cells sc_i and sc_j, Constraint (12) is reduced to

$$x_i + w_i \le x_j + M \cdot \frac{y_i \ominus y_j}{(k_i + k_j)R_h}, \text{ if } x_i^0 \le x_j^0. \tag{14}$$

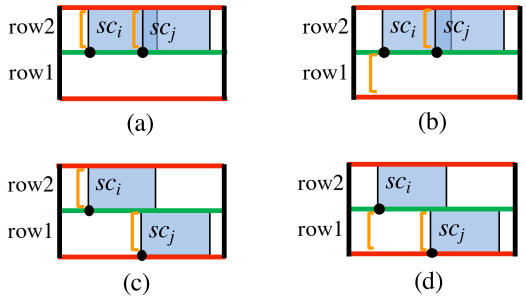


Figure 5: Four possible structures for two adjacent sub-cells i and j with $VMI_i \cap VMI_j \neq \emptyset$.

Finally, in Constraint (14), M is large and $\frac{y_i \ominus y_j}{(k_i+k_j)R_h}$ is a real number instead of an integer. If $y_i \ominus y_j > 0$, then $M\frac{y_i \ominus y_j}{(k_i+k_j)R_h}$ may be too large due to M, and then Constraint (14) may be invalid for resolving the horizontal overlap between sub-cells sc_i and sc_j. In addition, with Objective (2), x_i should not be far away from x_i^0. Hence, we set M as

$$M = \beta_{ij} \cdot (w_i + w_j), \tag{15}$$

where β_{ij} is a user-defined parameter, which is used to control the value of M.

Thus far, the mixed-cell-height standard-cell legalization problem is reformulated into a quadratic programming problem (QP):

$$\min_{x,y} \quad \sum_{sc_i \in SC} \frac{\alpha_i}{2r_i}((x_i - x_i^0)^2 + (y_i - y_i^0)^2) \tag{16}$$

$$\text{s.t.} \quad x_i + w_i \le x_j + \frac{\beta_{ij} \cdot (w_i + w_j)}{(k_i + k_j)R_h} \cdot (y_i \ominus y_j),$$
$$\forall sc_i, sc_j \in SC, \text{if } VMI_i \cap VMI_j \neq \emptyset,$$
$$sc_i, sc_j \text{ are adjacent, and } x_i^0 \le x_j^0; \tag{16a}$$

$$y_i \in \begin{cases} [y_i' - k_i^l R_h, y_i'], & \text{if } o_i \text{ is downward,} \\ [y_i', y_i' + k_i^u R_h], & \text{if } o_i \text{ is upward,} \end{cases}$$
$$x_i \ge 0, \ \forall sc_i \in SC; \tag{16b}$$
$$y_{i1} + (r_i - 1)R_h = y_{i2} + (r_i - 2)R_h = \cdots = y_{ir_i},$$
$$x_{i1} = x_{i2} = \cdots = x_{ir_i}, \ sc_{i1}, \cdots sc_{ir_i} \text{are from}$$
$$\text{the same cell.} \tag{16c}$$

3.3 Linear Complementarity Problem (LCP)

Since it is usually time-consuming to solve a large-scale quadratic programming problem with many inequality constraints, we convert the QP (Problem (16)) equivalently into an LCP, and solve the LCP by the modulus-based matrix splitting iteration method (MMSIM) [5]. To guarantee the convergence of MMSIM, it requires that the objective matrix is symmetric positive definite and the constraint matrix is of full row rank.

Let $x = (x_1, x_2, \cdots, x_n)^T$, $y = (y_1, y_2, \cdots, y_n)^T$, and $\mu = (\mu_i)_{2n} = \begin{bmatrix} x \\ y \end{bmatrix}$. Problem (16) can be rewritten as follows:

$$\min_{\mu} \quad \frac{1}{2}\mu^T Q \mu + p^T \mu \tag{17}$$
$$\text{s.t.} \quad A\mu \ge b; \tag{17a}$$
$$d \le y \le d^u; \tag{17b}$$
$$x \ge 0; \tag{17c}$$
$$E\mu = f, \tag{17d}$$

where Q is a diagonal matrix with its elements $q_{i,i} = q_{n+i,n+i} = \frac{\alpha_i}{r_i}$, $i = 1, 2, \cdots, n$; p is a vector with $p_i = -\frac{\alpha_i x_i^0}{r_i}$, $i = 1, 2, \cdots, n$; $p_i = -\frac{\alpha_i y_i^0}{r_{i-n}}$, $i = n+1, n+2, \cdots, 2n$; and A is the overlap constraint matrix with only four nonzero elements $1, -1, \frac{\beta_{ij}(w_i+w_j)}{(k_i+k_j)R_h}$, and $-\frac{\beta_{ij}(w_i+w_j)}{(k_i+k_j)R_h}$ in each row.

In order to obtain the uniform boundary constraint $\overline{\mu} \ge 0$, let $\overline{y} = y - d$, and $\overline{\mu} = (\overline{\mu}_i)_{2n} = \begin{bmatrix} x \\ \overline{y} \end{bmatrix}$. We have

$$\min_{\overline{\mu}} \quad \frac{1}{2}\overline{\mu}^T Q \overline{\mu} + \overline{p}^T \overline{\mu} \tag{18}$$
$$\text{s.t.} \quad A\overline{\mu} \ge \overline{b}; \tag{18a}$$
$$-I\overline{y} \ge d - d^u; \tag{18b}$$
$$\overline{\mu} \ge 0; \tag{18c}$$
$$E\overline{\mu} = \overline{f}, \tag{18d}$$

where I is an identity matrix, and vectors $\overline{p}$, $\overline{b}$, $\overline{f}$ are transformed from p, b, f, respectively.

To guarantee that the constraint matrix is of full row rank, we increase the number of variables by duplicating $\overline{y}$ to $\underline{y}$ with $\underline{y} = \overline{y}$. Then Constraint (18b) is replaced by $-I\underline{y} \ge d - d^u$, and $\underline{y} - \overline{y} = 0$. Let $E' \begin{bmatrix} \overline{y} \\ \underline{y} \end{bmatrix} = 0$ denote $\underline{y} - \overline{y} = 0$. In this way, Constraints (18a) and $-I\underline{y} \ge d - d^u$ compose a new system of inequality constraints, and Constraints (18d) and $\underline{y} - \overline{y} = 0$ form another new system of equality constraints.

Let $\widetilde{\mu} = \begin{bmatrix} x \\ \overline{y} \\ \underline{y} \end{bmatrix}$, $\widetilde{Q} = \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix}$, $\widetilde{p} = \begin{bmatrix} \overline{p} \\ 0 \end{bmatrix}$, $\widetilde{A} = \begin{bmatrix} A & 0 \\ 0 & -I \end{bmatrix}$, $\widetilde{b} = \begin{bmatrix} \overline{b} \\ d - d^u \end{bmatrix}$, $\widetilde{E} = \begin{bmatrix} E \\ E' \end{bmatrix}$, $\widetilde{f} = \begin{bmatrix} \overline{f} \\ 0 \end{bmatrix}$. Then we describe the standard form of the quadratic programming problem as follows:

$$\min_{\widetilde{\mu}} \quad \frac{1}{2}\widetilde{\mu}^T \widetilde{Q} \widetilde{\mu} + \widetilde{p}^T \widetilde{\mu} \tag{19}$$
$$\text{s.t.} \quad \widetilde{A}\widetilde{\mu} \ge \widetilde{b}; \tag{19a}$$
$$\widetilde{E}\widetilde{\mu} = \widetilde{f}; \tag{19b}$$
$$\widetilde{\mu} \ge 0. \tag{19c}$$

For Problem (19), we relax Constraint (19b) by putting it into the objective with a penalty parameter $\lambda > 0$. And a new QP is formulated as follows:

$$\min_{\widetilde{\mu}} \quad \frac{1}{2}\widetilde{\mu}^T (\widetilde{Q} + \lambda \widetilde{E}^T \widetilde{E})\widetilde{\mu} + (\widetilde{p}^T - \lambda \widetilde{f}^T \widetilde{E})\widetilde{\mu} \tag{20}$$
$$\text{s.t.} \quad \widetilde{A}\widetilde{\mu} \ge \widetilde{b}; \tag{20a}$$
$$\widetilde{\mu} \ge 0. \tag{20b}$$

For Problem (20), we have following two propositions:

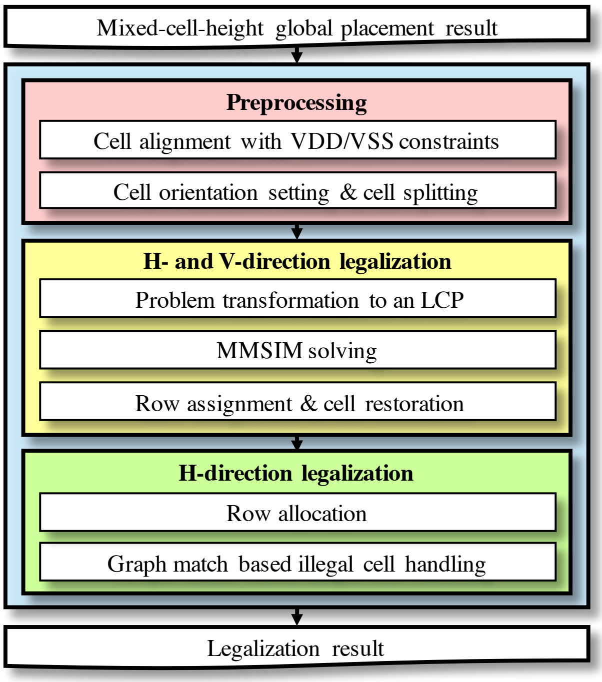


Figure 6: Our legalization framework.

Proposition 1. *In Problem (20), $\widetilde{Q}+\lambda\widetilde{E}^T\widetilde{E}$ is a symmetric positive definite matrix.*

Proposition 2. *In Problem (20), if $k_i^l = k_i^u = 1$ for all sub-cells, then matrix $\widetilde{A}$ is of full row rank.*

In this paper, k_i^l and k_i^u are set as 1 for each sub-cell. By Propositions (1) and (2), $\widetilde{\mu}$ is the global minimal solution of Problem (20) if and only if there exist vectors $r, u, v \geq 0$ such that the quadruple $(\widetilde{\mu}, r, u, v)$ satisfies the following KKT conditions [15]:

$$LCP(B,t): \; w = Bz + t \geq 0, \; z \geq 0, \; \text{and } z^T w = 0, \tag{21}$$

where $w = \begin{bmatrix} u \\ v \end{bmatrix}$, $B = \begin{bmatrix} \widetilde{Q}+\lambda\widetilde{E}^T\widetilde{E} & -\widetilde{A}^T \\ \widetilde{A} & 0 \end{bmatrix}$, $t = \begin{bmatrix} (\widetilde{p}-\lambda\widetilde{f}^T\widetilde{E}) \\ -\widetilde{b} \end{bmatrix}$, and $z = \begin{bmatrix} \widetilde{\mu} \\ r \end{bmatrix}$. Problem (21) is a linear complementarity problem. According to Theorem 1 of [5], we obtain the following theorem:

Theorem 1. *The solution of $LCP(B,t)$ (21) gives the optimal solution of QP (20), and vice versa.*

4 OUR LEGALIZATION FRAMEWORK

In this paper, we simultaneously minimize the average and maximum movements by a bi-directional (both horizontal- and vertical-direction) legalization method for mixed-cell-height standard cells. Our framework is summarized in Figure 6, which consists of three major steps: 1) preprocessing, 2) horizontal- and vertical-direction legalization, and 3) horizontal-direction legalization.

In the preprocessing step, all cells are first aligned to their nearest correct rows while meeting the VDD/VSS alignment constraints by shifting cells in the vertical direction. In addition, each cell is split into single-row height sub-cells. To determine the vertical moving interval VMI_i of y_i for each sub-cell, an orientation setting scheme is incorporated in Algorithm 1.

Algorithm 1 Sub-cell Orientation Setting

Input: a circuit with all sub-cells aligned to the correct rows;
Output: orientations o of all sub-cells in the vertical direction.

1: **for** $i = 0 : R_N$
2: calculate the average density ad_i^u of densities of rows $i+1$ to NR,
3: and the average density ad_i^d of densities of rows 0 to $i-1$;
4: $r^u = \max\{0, \min\{1, 0.5 + ad_i^u - ad_i^d\}\}$;
5: $W^u = w_{i,0}$, $o_{i,0} = 1$, $W^d = w_{i,1}$, $o_{i,1} = 0$;
6: **for** $j = 1 : R_i$
7: **if** $\frac{W^u}{W^d} \leq \frac{r^u}{1-r^u}$, **then** $W^u = W^u + w_{i,j}$, $o_{i,j} = 1$;
8: **else** $W^d = W^d + w_{i,j}$, $o_{i,j} = 0$;
9: **if** $(1-o_{i,j-1})o_{i,j} > 0$ and $y_{i,j-1}^0 > y_{i,j}^0$, **then**
10: $o_{i,j-1} = 1$, $o_{i,j} = 0$;
11: **if** $o_{i,j-1}(1-o_{i,j}) > 0$ and $y_{i,j-1}^0 < y_{i,j}^0$, **then**
12: $o_{i,j-1} = 0$, $o_{i,j} = 1$;
13: **Return** o.

In Algorithm 1, R_N is the number of rows in a circuit, and R_i is the number of sub-cells in row i. $o_{i,j}$ is the orientation of the j-th sub-cell $sc_{i,j}$ in row i. If $o_{i,j} = 1$, the orientation of sub-cell $sc_{i,j}$ is upward; it is downward, otherwise. $w_{i,j}$ and $y_{i,j}^0$ are the width and initial y-coordinate of sub-cell $sc_{i,j}$, respectively. In Lines 2–4, the ratio of the number of upward cells r^u to that of downward cells $1-r^u$ are calculated. Lines 7–12 determine the orientation $o_{i,j}$ of the j-th sub-cell in Row R_i.

In the horizontal- and vertical-direction legalization step, we first split cells into sub-cells. Then as described in Section 3, we model the mixed-cell-height standard-cell legalization problem as an $LCP(B,t)$ problem, which simultaneously considers cell moving in both of the horizontal and vertical directions. Then, we apply the MMSIM [5] to solve the $LCP(B,t)$ in Problem (21). If matrix B is positive definite, the convergence of the MMSIM for $LCP(B,t)$ holds [3]. However, the matrix B could be a positive semi-definite asymmetric matrix. As a result, the convergence of the MMSIM for solving $LCP(B,t)$ cannot be guaranteed. In this work, $LCP(B,t)$ in Problem (21) is resolved by an asymptotic modulus-based approach. The matrix B in $LCP(B,t)$ (Problem 21) is approximated by a positive definite matrix $B(\varepsilon) = B + \varepsilon I$, where I is an identify matrix and ε is a small constant. Consequently, the convergence of the MMSIM for solving $LCP(B(\varepsilon), t)$ can be guaranteed.

Our MMSIM-based algorithm is summarized in Algorithm 2. In Algorithm 2, Lines 1–9 give the MMSIM iterations, and each sub-cell is aligned to a correct row in Lines 10–16. In Line 5, we choose the splitting matrices $M(\varepsilon)$ and N with $B(\varepsilon) = M(\varepsilon) - N$ as follows:

$$M = \begin{bmatrix} \frac{1}{\beta^*}(\widetilde{Q}+\lambda\widetilde{E}^T\widetilde{E}) & 0 \\ \widetilde{A} & \frac{1}{\theta^*}D \end{bmatrix}, \; M(\varepsilon) = M + \varepsilon I,$$
$$N = \begin{bmatrix} (\frac{1}{\beta^*}-1)(\widetilde{Q}+\lambda\widetilde{E}^T\widetilde{E}) & -\widetilde{A}^T \\ 0 & \frac{1}{\theta^*}D \end{bmatrix}, \tag{22}$$

where $D = tridiag(\widetilde{A}(\widetilde{Q}+\lambda\widetilde{E}^T\widetilde{E}+\varepsilon I_n)^{-1}\widetilde{A}^T)$, in which $(\widetilde{Q}+\lambda\widetilde{E}^T\widetilde{E}+\varepsilon I_n)^{-1}$ can efficiently be calculated similarly as in [5], and β^*, θ^* are two positive constants as in [5]. After the execution of Algorithm 2, all multi-row height cells are restored, and we have the following theorem:

Theorem 2. *The iteration sequence $\{z^{(l)}\}_{k=0}^{+\infty} \subset \mathbb{R}_+^n$ generated by Algorithm 2 converges to the unique solution $z^* \in \mathbb{R}_+^n$ of $LCP(B(\varepsilon), t)$ for any initial vector $s^{(0)} \in \mathbb{R}^n$.*

Algorithm 2 Horizontal- and Vertical-Direction Legalization

Input: matrices: $M(\varepsilon)$, N, $B(\varepsilon)$; vectors: t, $s^{(0)}$, $z^{(0)}$; parameters: γ, δ, σ, κ;
Output: horizontal- and vertical-direction legalization result.

$l = 0$;
do
  $x_i^{(l)} = z_i^{(l)}$, $y_i^{(l)} = z_{n+i}^{(l)} + d_i$, $i = 1, 2, \cdots, n$;
  $\alpha_i = \left(\frac{(x_i^{(l)} - x_i^0)^2 + (y_i^{(l)} - y_i^0)^2}{\frac{1}{n}\sum_{j=1}^{n}(|x_j^{(l)} - x_j^0| + |y_j^{(l)} - y_j^0|)} \right)^{\kappa}$;
  solve $(M(\varepsilon) + I)s^{(l+1)} = Ns^{(l)} + (I - B(\varepsilon))|s^{(l)}| - \gamma t$;
  $z^{(l+1)} = \frac{1}{\gamma}(|s^{(l+1)}| + s^{(l+1)})$;
  l++;
until $|z^{(l)} - z^{(l-1)}| < \delta$
obtain the coordinate (x, y) for each sub-cell by $z^{(l)}$;
$i = 0$;
do
  if $\frac{|y_i - y_i'|}{R_h} < \sigma$, **then** $y_i = y_i'$;
  if $\frac{y_i - (y_i' - kR_h)}{R_h} < 1 - \sigma$, **then** $y_i = y_i' - kR_h$;
  if $\frac{(y_i' + kR_h) - y_i}{R_h} < 1 - \sigma$, **then** $y_i = y_i' + kR_h$;
  i++;
until $i \geq n$
Return x, y.

After the horizontal- and vertical-direction legalization, there could still exist overlaps due to the row assignment by rounding. In addition, each cell must be aligned to a placement site. In the horizontal-direction legalization step, we first align each cell to its nearest placement site. After that, if cell c_i does not overlap with any other cell, the location of c_i will be fixed; Meanwhile, a cell is marked as an illegal cell if this cell is overlapped with any other cell or out of the right boundary. For every illegal cell, we search all possible empty spaces. Then, a bipartite graph to model the illegal cells and empty spaces is constructed, and the Kuhn-Munkres algorithm [14] is applied to find a best matching for illegal cells and empty spaces, which runs in $O(n_{ill}^3)$ time, where n_{ill} is the number of illegal cells. After the above operations, all cells are placed in the chip region legally.

5 EXPERIMENTAL RESULTS

We implemented our analytical mixed-cell-height legalization algorithm in the C++ programming language. To evaluate the effectiveness of our proposed algorithm, we compared with the method in DAC'17 [5]. The benchmarks are modified from the ICCAD-2017 CAD Contest on Multi-Deck Standard-Cell Legalization [9] by omitting the fence-region constraints and the soft constraints. In our algorithm, the parameter λ in Problem (20) affects the mismatch distances of the sub-cells of a multi-row-height cell. Since the MMSIM is convergent, there will be theoretically no mismatch distance for each multi-row-height cell if the value of λ is large enough. Therefore, the parameter λ was set as 500. According to the works [3, 5], in Algorithm 2, β^* and θ^* in the splitting matrices M and N were both set as 0.5, and γ, σ, and κ were set as 1, 0.4, and 1, respectively, where the convergence of the MMSIM can be guaranteed. With the binaries provided by the authors of [5], all the experiments were run on the same PC with a 2.7GHz CPU and 16GB memory.

Table 1 lists the statistics of the benchmarks and the legalization results. In this table, for all benchmarks, "#Cell" gives the numbers of total standard cells, "#Single" the numbers of total single-row height standard cells, "#M" the numbers of total macro cells, "Den.(%)" the cell densities of circuits, "ΔHPWL(%)" the HPWL increases from the corresponding global placement results, "Avg. Move. (sites)" the average cell movements measured in the number of placement site width, "Max. Move. (sites)" the maximum cell movements measured in the number of placement site width, and "CPU(s)" the runtimes of all compared algorithms.

Table 1: Experimental results.

Benchmarks	#Cell	#Single	#M	Den.(%)
des_perf_1	112644	112644	0	90.59
des_perf_a_md1	108288	103589	4	55.05
des_perf_a_md2	108288	105030	4	55.86
des_perf_b_md1	112679	106702	0	54.98
des_perf_b_md2	112679	101908	0	64.69
edit_dist_1_md1	130661	118005	0	67.47
edit_dist_a_md2	127414	115066	6	59.42
edit_dist_a_md3	127414	119616	6	56.92
fft_2_md2	32281	28930	0	83.12
fft_a_md2	30625	27431	6	32.41
fft_a_md3	30625	28609	6	31.24
pci_bridge32_a_md1	29533	26680	4	49.57
pci_bridge32_a_md2	29533	25239	4	57.69
pci_bridge32_b_md1	28914	26134	6	26.47
pci_bridge32_b_md2	28914	28038	6	18.20
pci_bridge32_b_md3	28914	27452	6	22.13

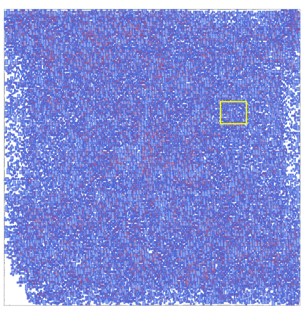
The experimental results are reported in Table 2. In the table, "DAC'17", "MIQP" and "LCP" represent the results of the algorithm in [5], the results obtained by solving our mixed integer quadratic programming (MIQP), and the results obtained by solving our linear complementarity problem (LCP), respectively. CPLEX [1] was chosen as our MIQP solver. Compared with the work "DAC'17", our LCP based algorithm achieves 32% shorter ΔHPWL, 14% smaller average cell movement, and 61% smaller maximum cell movement. The reason is that the work [5] resolves overlaps only in the horizontal direction, and if some overlaps among cells cannot be resolved in a row, they are marked and moved into empty positions of a layout in the illegal cell handling step. In contrast, our algorithm allows cells moving in both of the horizontal and vertical directions, and if some overlaps among cells cannot be resolved in a row, these cells are automatically moved into other adjacent rows. This horizontal- and vertical-direction legalization model can reduce not only the average cell movement, but also the maximum cell movement. Since the number of variables in our model is larger than that in [5], our average runtime is longer than that of DAC'17, yet still reasonably. Compared with the results in Columns "LCP", the MIQP based legalization algorithm achieves 2% shorter ΔHPWL, 2% smaller average cell movement. However, our LCP based algorithm can achieve an average speedup of 158× over the MIQP based algorithm.

This comparison further validates the efficiency and effectiveness of our LCP based analytical legalization algorithm.

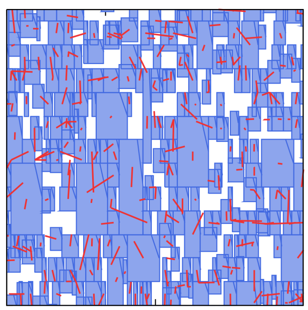
Figures 7(a) and 7(b) show the respective final layout and a partial layout of the benchmark "fft_2_md2" generated by our LCP based legalization algorithm.

Table 2: Experimental results.

Benchmarks	ΔHPWL(%)			Avg. Move. (sites)			Max. Move. (sites)			CPU(s)		
	DAC'17	MIQP	LCP	DAC'17	MIQP	LCP	DAC'17	MIQP	LCP	DAC'17	MIQP	LCP
des_perf_1	16.21	6.19	6.66	10.86	6.65	6.97	200.82	48.67	48.95	11.23	1672.24	11.75
des_perf_a_md1	3.27	2.47	2.48	6.71	5.93	5.94	607.30	607.30	607.30	2.30	917.02	2.79
des_perf_a_md2	3.35	2.52	2.51	6.77	5.90	5.93	403.86	403.86	403.86	2.19	839.70	6.82
des_perf_b_md1	1.75	1.50	1.52	5.17	4.75	4.77	79.34	51.14	38.45	2.01	905.74	3.64
des_perf_b_md2	2.05	1.68	1.72	5.74	5.22	5.25	198.74	54.75	39.76	2.31	991.30	3.12
edit_dist_1_md1	1.47	1.41	1.39	6.22	5.73	5.79	109.34	107.64	95.45	3.49	1300.40	5.19
edit_dist_a_md2	1.17	1.01	1.01	6.02	5.51	5.51	164.00	164.00	164.00	2.59	1313.57	2.24
edit_dist_a_md3	2.69	1.39	1.48	9.11	6.77	7.08	233.00	233.00	233.00	5.91	1357.08	15.68
fft_2_md2	11.21	8.52	8.78	8.84	7.44	7.54	102.94	62.69	73.60	0.70	69.52	2.89
fft_a_md2	0.98	0.95	0.95	5.03	4.86	4.86	345.50	345.50	345.50	0.69	57.03	0.60
fft_a_md3	1.08	1.07	1.08	4.73	4.54	4.55	109.62	109.62	109.62	0.63	55.56	0.40
pci_bridge32_a_md1	3.61	3.16	3.38	6.01	5.53	5.64	72.48	63.76	63.76	0.61	44.38	2.29
pci_bridge32_a_md2	8.33	4.26	4.38	9.43	7.01	7.14	186.08	121.35	121.35	0.53	54.57	3.34
pci_bridge32_b_md1	2.55	2.23	2.26	6.35	5.94	6.01	322.71	332.71	332.71	0.52	44.03	0.70
pci_bridge32_b_md2	2.80	2.54	2.53	5.92	5.52	5.53	640.12	430.04	430.04	0.50	42.11	0.66
pci_bridge32_b_md3	3.63	3.17	3.17	6.74	6.10	6.10	398.57	398.57	398.57	0.51	43.26	1.58
N.Average	1.32	0.98	1.00	1.14	0.98	1.00	1.61	1.02	1.00	0.67	158.25	1.00



(a)



(b)

Figure 7: (a) Legalization result of the benchmark "fft_2_md2" from our algorithm. Cells are in blue, and movement in red. (b) The partial layout of the region marked in (a).

6 CONCLUSION

In this paper, we have considered both the average and the maximum cell movements for the mixed-cell-height standard-cell legalization problem. By analyzing and remodeling the objective function and constraints, we formulated the mixed-cell-height standard-cell legalization problem as an MIQP, which considers not only the average cell movement, but also the maximum movement, the sub-maximum movement, the third maximum movement, etc. Then, the MIQP was relaxed to a QP, which allows cells spreading in both of the horizontal and vertical directions. By substituting and duplicating variables, the QP was further converted to an equivalent LCP. Then the MMSIM was used to solve the LCP efficiently. Experimental results have shown that our analytical legalization method is effective in reducing wirelength and the average and maximum cell movements.

REFERENCES

[1] CPLEX. (2018 May) [Online]. Available: http://www-01.ibm.com.

[2] S. H. Baek, H. Y. Kim, Y. K. Lee, D. Y. Jin, S. C. Park, and J. D. Cho. Ultra high density standard cell library using multi-height cell structure. In *Proceedings of SPIE 7268*, pages 72680C–72680C, 2008.

[3] Z. Z. Bai. Modulus-based matrix splitting iteration methods for linear complementarity problems. *Numerical Linear Algebra with Applications*, 17(6):917–933, December 2010.

[4] U. Brenner. Bonnplace legalization: Minimizing movement by iterative augmentation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 32(8):1215–1227, 2013.

[5] J. Chen, Z. Zhu, W. Zhu, and Y.-W. Chang. Toward optimal legalization for mixed-cell-height circuit designs. In *Proceedings of ACM/IEEE Design Automation Conference*, 2017.

[6] M. Cho, H. Ren, H. Xiang, and R. Puri. History-based VLSI legalization using network flow. In *Proceedings of ACM/IEEE Design Automation Conference*, pages 286–291, 2010.

[7] W.-K. Chow, C.-W. Pui, and E. F. Y. Young. Legalization algorithm for multiple-row height standard cell design. In *Proceedings of ACM/IEEE Design Automation Conference*, 2016.

[8] N. K. Darav, I. S. Bustany, A. Kennings, and L. Behjat. A fast, robust network flow-based standard-cell legalization method for minimizing maximum movement. In *Proceedings of ACM International Symposium on Physical Design*, 2017.

[9] N. K. Darav, I. S. Bustany, A. Kennings, and R. Mamidi. ICCAD-2017 CAD Contest in multi-deck standard cell legalization and benchmarks suite. In *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, 2017.

[10] N. K. Darav, A. Kennings, A. F. Tabrizi, D. Westwick, and L. Behjat. Eh?placer: A high-performance modern technology-driven placer. *ACM Transactions on Design Automation of Electronic Systems*, 21(3):37:1–37:27, April 2016.

[11] S. Dobre, A. B. Kahng, and J. Li. Mixed cell-height implementation for improved design quality in advanced nodes. In *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, pages 854–860, 2015.

[12] C.-Y. Hung, P.-Y. Chou, and W.-K. Mak. Mixed-cell-height standard cell placement legalization. In *Proceedings of Great Lakes Symposium on VLSI*, 2017.

[13] A. B. Kahng, I. L. Markov, and S. Reda. On legalization of row-based placements. In *Proceedings of Great Lakes Symposium on VLSI*, pages 214–219, 2004.

[14] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, March 1957.

[15] J. Nocedal and S. J. Wright. *Numerical optimization*. New York: Springer, 2006.

[16] C.-H. Wang, Y.-Y. Wu, J. Chen, Y.-W. Chang, S.-Y. Kuo, W. Zhu, and G. Fan. An effective legalization algorithm for mixed-cell-height standard cells. In *Proceedings of IEEE/ACM Asia and South Pacific Design Automation Conference*, 2017.

[17] G. Wu and C. Chu. Detailed placement algorithm for VLSI design with double-row height standard cells. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 35(9):1569–1573, 2015.