

A Method of Analog IC Placement with Common Centroid Constraints

Keitaro Ue

Kunihiro Fujiyoshi

Department of Electrical and Electronic Engineering
 Tokyo University of Agriculture and Technology
 2-24-16 Nakachou, Koganei-shi, Tokyo 184-8588, Japan
 keitaro@fjlab.ei.tuat.ac.jp, fujiyosi@cc.tuat.ac.jp

Abstract—To improve the immunity against process gradients, a *common centroid constraint*, in which every pair of capacitors which has been derived by dividing some original capacitors into two halves should be placed symmetrically with respect to a common centroid, is widely used. Xiao et al. proposed a method to obtain a placement satisfying the common centroid constraints, but this method has a defect. In this paper, we propose a decoding algorithm to obtain a placement which satisfies common centroid constraints.

I. INTRODUCTION

Recently most of integrated circuit chips are made from monolithic IC, which can be manufactured at low cost. In monolithic IC, absolute error which is proportion of the difference of actual value to the designed value of each element is large (several tens of percentages), but the maximum difference of the absolute error of devices on a chip, is small (a few percentages). So it is suitable to build circuits such as A/D, D/A converters and switched-capacitor filters. But if the accuracy of these circuits is required to be higher, such a small mismatch of capacitors or resistors still becomes a critical issue. The causes of the mismatch can be divided into two categories: systematic mismatch and random mismatch [1, 2]. One of the frequent systematic mismatch is caused by process gradients, whose effect can be represented by the following linear model [3]:

$$C_{xy} = C + \alpha x + \beta y,$$

where C_{xy} is the capacitance of the unit capacitor in the coordinates x and y , C is the ideal value of the unit capacitance, and α and β are the process gradient components in x and y directions, respectively.

Therefore to improve the immunity against process gradients, a *common centroid constraint*, in which every pair of capacitors which has been derived by dividing some original capacitors into two halves should be placed symmetrically with respect to a common centroid, is widely used.

In order to obtain placement which satisfies common centroid constraint, some methods are proposed [4, 5, 6,

7, 8]. Strasser et al. proposed a method using B*-tree [4], but this method is deterministic. P. H. Lin et al. proposed a thermal-driven common centroid placement algorithm [5], and C. W. Lin et al. proposed a method to obtain placement considering random mismatches [6]. Both methods [5, 6] can handle unit-capacitors only.

Two methods using sequence-pair (seq-pair) [9], which is a topological representation of rectangular block placement, are proposed [7, 8]. Ma et al. proposed a method using *Center-based Corner Block List (C-CBL)* [7], which is an extension of CBL [10]. They regard each common centroid group as a rectangular super-block, and search whole placements by seq-pair, so area obtained by the method may be redundantly large. Xiao et al. proposed a method which uses seq-pair to obtain placement satisfying common centroid constraint and searches seq-pairs which satisfy some constraints [8]. But their method has a defect that some blocks may be overlapped (Fig.1(e)).

In this paper, we introduce a method proposed in [8] and a defect of the method. And we propose a decoding algorithm of seq-pair to obtain a placement which satisfies common centroid constraints, without overlapping. The proposed decoding algorithm takes $O(n^2)$ time. The effectiveness of the method will be examined by computational experiments.

II. PROBLEM DEFINITION

A *common centroid constraint* gives a constraint, in which every pair of cells in the same given group should be placed symmetrically with respect to a common centroid of the group. These groups are called *common centroid groups*. A common centroid group consists of some pairs of cells and at most one self-symmetry cell, and the center of the self-symmetry cell conforms to the common centroid of the group. A common centroid group is represented by a set of cell pairs in parentheses and at most one self-symmetry cell such as $\{(a_1, a_2), b_s, (c_1, c_2)\}$.

In this paper, it is supposed that input seq-pair satisfies group clustered placement condition and common centroid condition, which are defined in section III., and cells in each common centroid group should be placed as close as possible to the common centroid. a_x denotes rel-

ative x -coordinate of left side of cell a , and a_y denotes relative y -coordinate of bottom side of cell a . $a.height$ and $a.width$ denote height and width of cell a respectively.

III. METHOD PROPOSED IN [8]

A method proposed by Xiao et al. searches only seq-pairs satisfying *group clustered placement condition*, which is the same as *separate constraint* defined in [12], and *common centroid condition*.

common centroid condition:

Suppose a seq-pair $S = (\Gamma_+; \Gamma_-)$ which includes common centroid group \mathcal{G}^i and $(\Gamma_{\mathcal{G}^i+}; \Gamma_{\mathcal{G}^i-})$ which is a sub-seq-pair obtained by extracting just the cells in \mathcal{G}^i from S . If $\Gamma_{\mathcal{G}^i+}$ and $\Gamma_{\mathcal{G}^i-}$ satisfy the following conditions, seq-pair S satisfies common centroid condition.

$$\begin{aligned}\Gamma_{\mathcal{G}^i+} &= sym(rev(\Gamma_{\mathcal{G}^i+})) \\ \Gamma_{\mathcal{G}^i-} &= sym(rev(\Gamma_{\mathcal{G}^i-})),\end{aligned}$$

where $rev(s)$ is a string obtained by reversing a string s and $sym(s)$ is a string obtained by replacing a string s by their symmetric counterparts.

The method in [8] has a defect that some cells may be overlapped. An example is as follows.

Suppose a seq-pair $S = (a_1eb_s a_2c_2gd_s c_1f; ea_1gc_1d_sfb_s a_2c_2)$ is given, and common centroid groups are $\mathcal{G}^1\{(a_1, a_2), b_s\}$, $\mathcal{G}^2\{(c_1, c_2), d_s\}$. The size of a_1 , a_2 and b_s is 1×1 , the size of c_1 , c_2 and d_s is 2×1 , the size of e and g is 2×2 , the size of f is 1×1.5 .

First, construct constraint graphs from input seq-pair S and calculate coordinates of each cell by the longest path length from source to the corresponding vertex (Fig.1(a),(b)). Then cells in \mathcal{G}^1 don't satisfy common centroid constraint. To satisfy common centroid constraint, calculate potential displacement δ which a_2 should be shifted by (Algorithm 1 AdjustingHorizontalSymmetry in [8]). And add the δ to weights of all out-going edges of a_2 (Fig.1(c),(d)). Finally, as cells in \mathcal{G}^1 and \mathcal{G}^2 satisfy common centroid constraint, re-calculate coordinates of cells in \mathcal{G}^1 and \mathcal{G}^2 . Then resultant placement is shown in Fig.1(e).

IV. PROPOSED METHOD

In symmetric placement, to determine the relative coordinates of pairs of cells which should be placed symmetrically with respect to a symmetrical axis is possible by using longest path algorithm on a constraint graph, if these cells should be placed as close as possible to the axis [11]. For example, suppose a pair of cells (a_1 and a_2) in a symmetric group is constrained horizontally by seq-pair, and a_1 is constrained to be left of a_2 . The relative x -coordinates of a_1 and a_2 satisfy the following relations:

$$\begin{aligned}subject\ to\ & a_1.x + a_1.width + a_2.x = 0 \\ & a_2.x \geq 0 \\ minimize\ & a_2.x,\end{aligned}$$

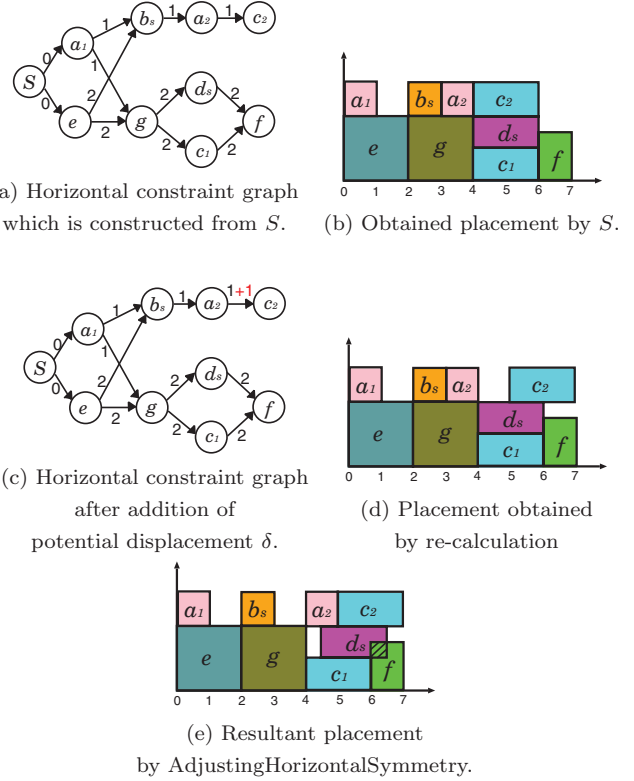


Fig. 1. Example of *Algorithm AdjustingHorizontalSymmetry* proposed in [8], which results in the overlap of cell d_s and f : common centroid groups $\mathcal{G}^1\{(a_1, a_2), b_s\}$, $\mathcal{G}^2\{(c_1, c_2), d_s\}$, asymmetry cells $\{e, f, g\}$, seq-pair $S = (a_1eb_s a_2c_2gd_s c_1f; ea_1gc_1d_sfb_s a_2c_2)$

where relative x -coordinate of the symmetric axis is zero. So these relative x -coordinates are not difficult to determine. And relative y -coordinates are not difficult to determine because the only additional constraint is to align relative y -coordinates of cells in each pair.

In common centroid placement, if a_1 is constrained to be left of a_2 , the relative x -coordinates of these cells are not difficult to determine by the same reason. But relative y -coordinates of these cells are difficult to determine, because the only relation of the relative y -coordinates of cells in the pair is

$$a_1.y + a_1.height + a_2.y = 0.$$

A. fundamental idea

Cells in common centroid group \mathcal{G}^i can be classified into R_i , L_i , A_i , B_i and S_i . Each cell in R_i is constrained to be right of a pair to the cell, which is in L_i , by seq-pair. Similarly each cell in A_i is constrained to be above a pair to the cell, which is in B_i . S_i is the set of self-symmetry cells. Note that $|S_i| \leq 1$. For each common centroid group, we partition the region into four quadrants and place cells in R_i in the first quadrant. Similarly cells in A_i are placed in the second quadrant, and cells in L_i are

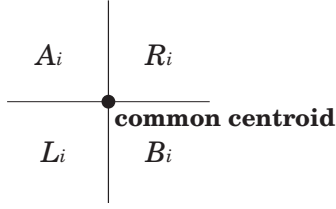


Fig. 2. Four regions in which cells in each subset R_i, A_i, L_i, B_i are placed by fundamental idea

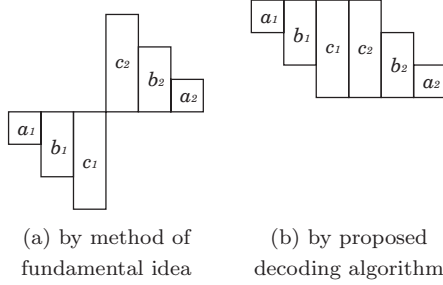


Fig. 3. Obtained placements by by method of fundamental idea and proposed method: seq-pair $(a_1b_1c_1c_2b_2a_2; a_1b_1c_1c_2b_2a_2)$, common centroid group $\mathcal{G}^1 \{(a_1, a_2), (b_1, b_2), (c_1, c_2)\}$: $|A_i| = |B_i| = |S_i| = 0$

placed in the third quadrant, cells in B_i are placed in the fourth quadrant (Fig.2).

- The fundamental idea has the following weak points:
- (1) If $|S_i| \neq 0$, there exist dead space in each quadrant around the only element of S_i as shown in hatched region on Fig.5(j).
 - (2) If $|S_i| = |A_i| = |B_i| = 0$, vertical distance between cells in R_i and cells in L_i is redundantly large as shown in Fig.3(a).
 - (3) If $|S_i| = |R_i| = |L_i| = 0$, horizontal distance between cells in A_i and cells in B_i are redundantly large.

B. proposed decoding algorithm

To overcome the above weak point (1), we change the position of the following four regions in which cells in each subset R_i, A_i, L_i, B_i are placed (Fig.4). If $|S_i| = \{s\}$,

$$\begin{aligned}
 \text{if } a \in R_i & : \begin{cases} \frac{s.width}{2} \leq a.x \\ -\frac{s.height}{2} \leq a.y \end{cases} \\
 \text{if } a \in A_i & : \begin{cases} \frac{s.width}{2} \geq a.x + a.width \\ \frac{s.height}{2} \leq a.y \end{cases} \\
 \text{if } a \in L_i & : \begin{cases} -\frac{s.width}{2} \geq a.x + a.width \\ \frac{s.height}{2} \geq a.y + a.height \end{cases} \\
 \text{if } a \in B_i & : \begin{cases} -\frac{s.width}{2} \leq a.x \\ -\frac{s.height}{2} \geq a.y + a.height \end{cases}
 \end{aligned}$$

To overcome the weak point (2), if $|S_i| = |A_i| = |B_i| = 0$, we change the relative y -coordinates of cells in L_i , and to overcome the weak point (3), if $|S_i| = |R_i| = |L_i| = 0$, we change the relative x -coordinates of cells in A_i . Proposed decoding algorithm is as follows.

input: set of cells, cell sizes, common centroid constraints, seq-pair.

step1: Based on the horizontal (left of) constraints imposed by the seq-pair, a directed weighted graph $G_h(V, E)$ (V : vertex set, E : edge set), called *horizontal constraint graph* is constructed as follows [9].

- V consists of source sv and vertices labeled with the corresponding names of all cells.
- E consists of (sv, p) for every cell p , and (p, q) if and only if p is constrained to be left of q by the seq-pair.
- Length of edge (p, q) is the width of the cell p .

Similarly, the *vertical constraint graph* G_v is constructed using vertical (below) constraints and the height of each cell.

step2: For each common centroid group \mathcal{G}^i , carry out the following steps.

If $|S_i| \neq 0$, suppose the only self-symmetry cell in \mathcal{G}^i is denoted as s . If $|S_i| = 0$, $s.width = s.height = 0$.

step2.1: Make induced subgraph G_h^i of G_h , induced by vertices in R_i, B_i and S_i . Add source sv and edge from sv to every vertex to G_h^i . For vertex c ,

$$weight(sv, c) = \begin{cases} 0 : (\text{if } c \in R_i) \\ -\frac{s.width}{2} : (\text{if } c \in (B_i \cup S_i)) \end{cases}$$

Calculate the longest path length from sv to each vertex and regard them as relative x -coordinates of left side of all cells in \mathcal{G}^i .

step2.2: Make induced subgraph G_v^i of G_v , induced by vertices in R_i, A_i and S_i , and add source sv and edges (sv, c) , similar to step2.1. For vertex c ,

$$weight(sv, c) = \begin{cases} 0 : (\text{if } c \in A_i) \\ -\frac{s.height}{2} : (\text{if } c \in (R_i \cup S_i)) \end{cases}$$

Calculate the longest path length from sv to each vertex and regard them as relative y -coordinates of bottom side of all cells in \mathcal{G}^i .

step2.3: If $|S_i| = |A_i| = |B_i| = 0$: Change the relative y -coordinates of cells in L_i so that the minimum relative y -coordinate of cells in R_i is equal to the minimum relative y -coordinate of cells whose relative x -coordinate of right side is equal to zero in L_i . (See Fig.3(b)).

step2.4: If $|S_i| = |R_i| = |L_i| = 0$: Change the relative x -coordinates of cells in A_i so that the minimum relative x -coordinate of cells in B_i is equal to the minimum relative x -coordinate of cells whose relative y -coordinate of bottom side is equal to zero in A_i .

Regard a set of cells in each symmetry group as one convex block.

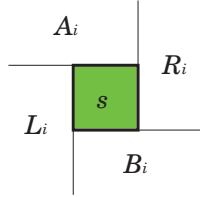


Fig. 4. Four regions in which cells in each subset R_i, A_i, L_i, B_i are placed by proposed decoding algorithm: $\{s\} = S_i$

step3: Coordinates of all cells are determined by convex block packing algorithm, proposed in [12]. Detailed operations are as follows.

step3.1: Adjust weight of edges in G_h using relative coordinates obtained in step2. Weight of edge (a, b) is adjusted to

$$a.width + a.x - b.x,$$

where $a.x$ is relative x -coordinate of a in \mathcal{G}^i if $a \in \mathcal{G}^i$, and $a.x = 0$ if a is an asymmetry cell, $b.x$ is similarly.

Similarly adjust weight of edges in G_v .

step3.2: Merge all the vertices in \mathcal{G}^i into one “super-vertex”, and calculate the longest path length from source to all vertices in G_h and G_v . And let these longest path lengths be x and y coordinates of corresponding cells or convex blocks. ■

The time complexity of step1 and step2 is obviously $O(n^2)$, and that of step3 is $O(n^2)$, which is shown in [12]. So the total time complexity of this proposed decoding algorithm is $O(n^2)$.

For example, suppose a seq-pair $(b_1 a_1 c_s a_2 b_2 d; b_2 d a_1 c_s a_2 b_1)$ which satisfies common centroid condition and group clustered placement condition is given. And common centroid group is $\mathcal{G}^1 \{(a_1, a_2), (b_1, b_2), c_s\}$. The size of a_1, a_2 and c_s is 1×1 , the size of b_1, b_2 and d is 1×2 .

step1: constraint graphs G_h, G_v of the input seq-pair are shown in Fig.5(a), (b).

step2.1: induced subgraph G_h^1 of G_h induced by cells in the symmetry group \mathcal{G}^1 is shown in Fig.5(c). step2.2: induced subgraph G_v^1 of G_v induced by cells is shown in Fig.5(d).

step3.1: for each constraint graph, widths of edges are adjusted using relative coordinates obtained in step2 (Fig.5(e),(f)). step3.2: merge all the vertices in \mathcal{G}^1 into one “super-vertex” as shown in Fig.5(g),(h). Calculate the longest path length from source to all vertices in G_h'' , and let these values be x -coordinates of corresponding cells or convex blocks. Similarly calculate the longest path length from source to all vertices in G_v'' , and let these values be y -coordinates of that (Fig.5(i)). For comparison, placement which is obtained from the same input by method of fundamental idea is shown in Fig.5(j).

V. EXPERIMENTAL RESULT

In order to confirm effectiveness of the proposed decoding algorithm, we implemented the decoding algorithm and a method of our fundamental idea with Simulated Annealing (SA) in C language, and carried out experiments by Athlon 2.7GHz for two data used in [4] and [8], which are “biasynth_2p4g” and “lnamixbias_2p4g”. “biasynth_2p4g” consists of 65 cells with 3 common centroid groups (4 pairs, 6 pairs, and 2 pairs and 1 self symmetric cell). “lnamixbias_2p4g” consists of 110 cells with five common centroid groups (8 pairs, 3 pairs, 3 pairs, 6 pairs, and 2 pairs). Table I shows the experimental comparisons of results of the proposed method with [8]. “Area ratio” is defined by the following equation:

$$Area\ ratio = \frac{(area\ of\ bounding\ box)}{(total\ area\ of\ all\ cells)}.$$

We cannot simply compare these results, because we considered all groups in “biasynth_2p4g” and “lnamixbias_2p4g” are common centroid groups, but [8] considered two groups in “biasynth_2p4g” and 3 groups in “lnamixbias_2p4g” are symmetry groups. The area of placement obtained by the proposed decoding algorithm is less than that by the method of fundamental idea, though the CPU time is almost similar. The packing results of “biasynth_2p4g” obtained by proposed decoding algorithm and that by the method of fundamental idea are shown in Fig.6 and Fig.8, and the packing results of “biasynth_2p4g” are shown in Fig.7 and Fig.9 respectively.

VI. CONCLUSION

In this paper, we proposed a decoding algorithm to obtain placement satisfying common centroid constraints and relative position constraints imposed by seq-pair, which satisfies both groups clustered placement condition and common centroid condition, in $O(n^2)$ time. And, in experiments, feasible placements satisfying common centroid constraints were obtained in about two hundred seconds.

REFERENCES

- [1] M. J. McNutt, S. LeMarquis, and J. L. Dunkley, “Systematic capacitance matching errors and corrective layout procedures”, Proc. IEEE JSSC, pp. 611–616, 1994.
- [2] J. B. Shyu, G. C. Temes and F. Krummenacher. “Random Error Effects in Matched MOS Capacitors and Current Sources”, Proc. IEEE J. of Solid-State Circuits, pp.948–955, 1984.
- [3] M. Kaneko, M. Masuda, and T. Hayashi: “A novel capacitor placement strategy in ascot: automatic layouter for switched-capacitor circuits”, Proc. IEEE ISCAS, pp.2094–2097, 1993.
- [4] M. Strasser, M. Eick, H. Graeb, U. Schlichtmann, and F. M. Johannes, “Deterministic analog circuit placement using hierarchically bounded enumeration and enhanced shape functions”, Proc. ICCAD, pp. 306–313, 2008.

TABLE I
EXPERIMENTAL COMPARISON OF PROPOSED DECODING ALGORITHM (ATHLON 2.7GHz) WITH METHOD PROPOSED IN [8](XEON 2.2GHz)

Design	#Cell	#Symmetry groups	method in [8]		fundamental idea		proposed decoding algorithm	
			Time[sec]	Area ratio[%]	Time[sec]	Area ratio[%]	Time[sec]	Area ratio[%]
biasynth_2p4g	65	8+12+5	18.2	103.91	152.0	110.43	141.6	108.88
Inamixbias_2p4g	110	16+6+6+12+4	58.7	104.31	229.0	116.56	222.4	112.39

- [5] P. H. Lin, H. Zhang, M. D. F. Wong, and Y. W. Chang, "Thermal-driven analog placement considering device matching", Proc. DAC, pp. 593–598, 2009.
- [6] C. W. Lin, J. M. Lin, Y. C. Chiu, C. P. Huang and S. J. Chang: "Common-Centroid Capacitor Placement Considering Systematic and Random Mismatches in Analog Integrated Circuits", Proc. DAC, pp.528–533, 2011.
- [7] Q. Ma and E. F. Y. Young, and K. P. Pun: "Analog Placement with Common Centroid Constraints", Proc. ICCAD, pp.579–585, 2007.
- [8] L.Xiao and E. F. Y. Young: "Analog Placement with Common Centroid and 1-D Symmetry Constraints", Proc. ASP-DAC, pp.353–360, 2009.
- [9] H. Murata, K. Fujiyoshi, S. Nakatake and Y. Kajitani: "Rectangle-packing-based module placement", Proc. IEEE IC-CAD, pp.472–479, 1995.
- [10] X. Hong, G. Huang, Y. Cai, J. Gu, S. Dong, C. K. Cheng, and J. Gu: "Corner Block List: An Effective and Efficient Topological Representation of Non-Slicing Floorplan", Proc. ICCAD, pp.8–13, 2000.
- [11] S. Koda, K. Fujiyoshi, "On Handling Cell Placement with Adjacent Symmetry Constraints for Analog IC Layout Design" IEICE Technical Report VLD2006-77, pp.31–36, 2006.
- [12] K. Wakata, H. Saito, K. Fujiyoshi, K. Sakanushi, T. Obata and C. Kodama: "An Improved Method of Convex Rectilinear Block Packing Based on Sequence-Pair", IEICE Trans, Fundamentals, vol.E86-A, no.12, pp.3148–3157, 2003.

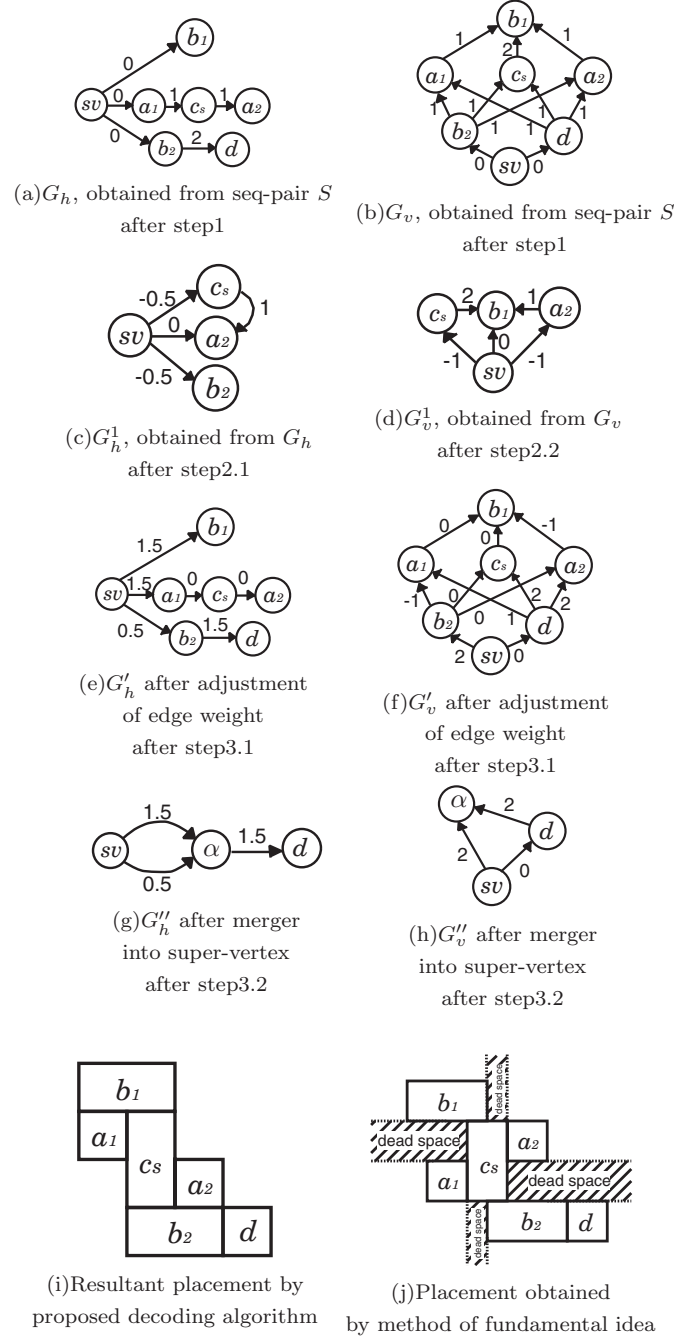


Fig. 5. An example of the decoding algorithm : common centroid group $\mathcal{G}^1\{(a_1, a_2), (b_1, b_2), c_s\}$, asymmetry cell d , seq-pair $S=(b_1 a_1 c_s a_2 b_2 d; b_2 d a_1 c_s a_2 b_1)$

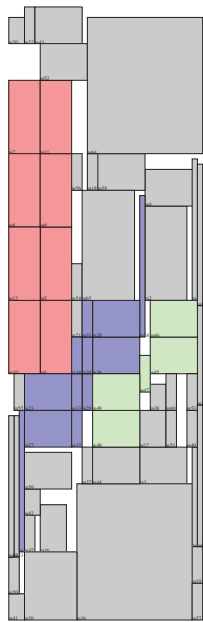


Fig. 6. Resultant placement of “biasynth_2p4g” obtained by method of fundamental idea (Time: 152.0[sec], Area ratio: 110.43[%])

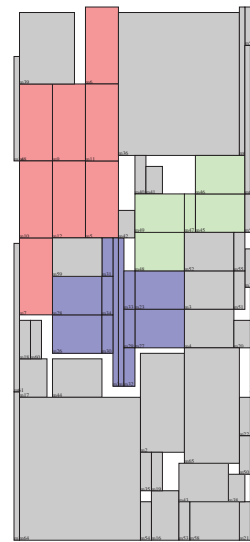


Fig. 8. Resultant placement of “biasynth_2p4g” obtained by proposed decoding algorithm (Time: 141.6[sec], Area ratio: 108.88[%])

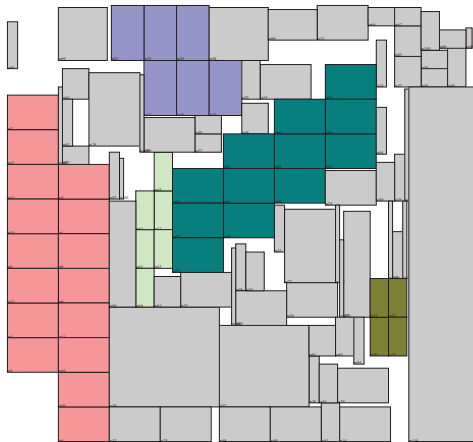


Fig. 7. Resultant placement of “lnamixbias_2p4g” obtained by method of fundamental idea (Time: 229.0[sec], Area ratio: 116.56[%])

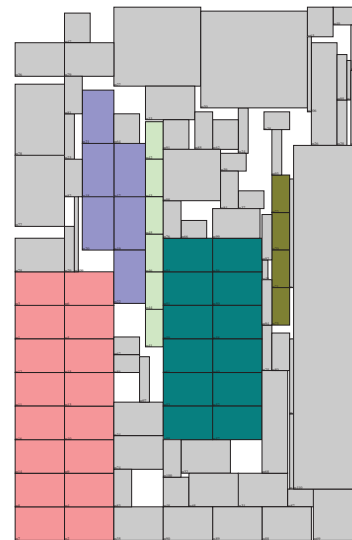


Fig. 9. Resultant placement of “lnamixbias_2p4g” obtained by proposed decoding algorithm (Time: 222.4[sec], Area ratio: 112.39[%])