Theoretical Computer Science 532 (2014) 40–50

Contents lists available at SciVerse ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

A simple optimal binary representation of mosaic floorplans and Baxter permutations

Bryan Dawei He

Department of Computer Science, California Institute of Technology, Pasadena, CA 91126, United States

ARTICLE INFO

Keywords:
Floorplan
Mosaic floorplan
Baxter permutation
Binary representation

ABSTRACT

Mosaic floorplans are rectangular structures subdivided into smaller rectangular sections and are widely used in VLSI circuit design. Baxter permutations are a set of permutations that have been shown to have a one-to-one correspondence to objects in the *Baxter combinatorial family*, which includes mosaic floorplans. An important problem in this area is to find short binary string representations of the set of n-block mosaic floorplans and Baxter permutations of length n. The best known representation is the *Quarter-State Sequence* which uses $4n$ bits. This paper introduces a simple binary representation of n-block mosaic floorplan using $3n-3$ bits. It has been shown that any binary representation of n-block mosaic floorplans must use at least $(3n-o(n))$ bits. Therefore, the representation presented in this paper is optimal (up to an additive lower order term).

1. Introduction

In this section, we introduce the definition of mosaic floorplans and Baxter permutations, describe their applications and previous work in this area, and state our main result.

1.1. Floorplans and mosaic floorplans

Definition 1. A *floorplan* is a rectangle subdivided into smaller rectangular subsections by horizontal and vertical line segments such that no four subsections meet at the same point.

The smaller rectangular subsections are called *blocks*. Fig. 1 shows three floorplans, each containing 9 blocks. Note that the horizontal and vertical line segments do not cross each other. They can only form *T-junctions* ($\vdash$, $\perp$, $\dashv$, and $\top$).

The definition of equivalent floorplans does not consider the size of the blocks of the floorplan. Instead, two floorplans are considered equivalent if and only if their corresponding blocks have the same relative position relationships. The formal definition is given below.

Definition 2. Let F_1 be a floorplan with R_1 as its set of blocks. Let F_2 be another floorplan with R_2 as its set of blocks. F_1 and F_2 are considered *equivalent floorplans* if and only if there is a one-to-one mapping $g : R_1 \to R_2$ such that the following conditions hold:

1. For any two blocks $r, r' \in R_1$, r and r' share a horizontal line segment as their common boundary with r above r' if and only if $g(r)$ and $g(r')$ share a horizontal line segment as their common boundary with $g(r)$ above $g(r')$.

E-mail address: bryanhe@caltech.edu.

0304-3975/$ – see front matter
http://dx.doi.org/10.1016/j.tcs.2013.05.007

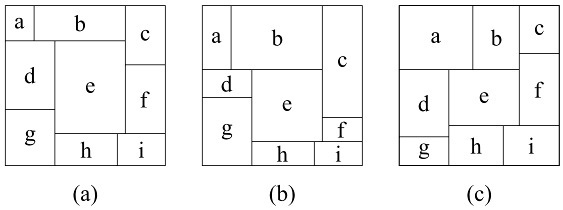

Fig. 1. Three example floorplans.

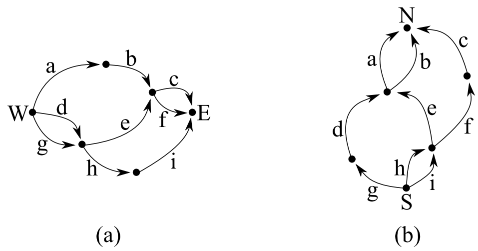

Fig. 2. The constraint graphs representing all three mosaic floorplans in Fig. 1. (a) is the horizontal constraint graph. (b) is the vertical constraint graph.

2. For any two blocks $r, r' \in R_1$, r and r' share a vertical line segment as their common boundary with r to the left of r' if and only if $g(r)$ and $g(r')$ share a vertical line segment as their common boundary with $g(r)$ to the left of $g(r')$.

In Fig. 1 (a) and (b) have the same number of blocks and the position relationships between their blocks are identical. Therefore, (a) and (b) are equivalent floorplans. However, (c) is not equivalent to either.

The objects of *mosaic floorplans* are the same as the objects of the floorplans. However, mosaic floorplans use a different definition of equivalence. Informally speaking, two mosaic floorplans are considered equivalent if and only if they can be converted to each other by sliding the horizontal and vertical line segments. The equivalence of the mosaic floorplans is formally defined by using the *horizontal constraint graph* and the *vertical constraint graph* [9]. The horizontal constraint graph describes the horizontal relationship between the vertical line segments of a floorplan. The vertical constraint graph describes the vertical relationship between the horizontal line segments of a floorplan. The formal definitions are given below.

Definition 3. Let F be a floorplan.

1. The *horizontal constraint graph* $G_H(F)$ of F is a directed graph. The vertex set of $G_H(F)$ 1-to-1 corresponds to the set of the vertical line segments of F. For two vertices u_1 and u_2 in $G_H(F)$, there is a directed edge $u_1 \to u_2$ if and only if there is a block b in F such that the vertical line segment v_1 corresponding to u_1 is on the left boundary of b and the vertical line segment v_2 corresponding to u_2 is on the right boundary of b.
2. The *vertical constraint graph* $G_V(F)$ of F is a directed graph. The vertex set of $G_V(F)$ 1-to-1 corresponds to the set of the horizontal line segments of F. For two vertices u_1 and u_2 in $G_H(F)$, there is a directed edge $u_1 \to u_2$ if and only if there is a block b in F such that the horizontal line segment h_1 corresponding to u_1 is on bottom boundary of b and the horizontal line segment h_2 corresponding to u_2 is on the top boundary of b.

The graphs in Fig. 2 are the constraint graphs of all three floorplans shown in Fig. 1 (a), (b), and (c). Note that the bottom (top, left, right, respectively) boundary of the floorplan is represented by the south (north, west, east, respectively) vertex labeled by S (N, W, E, respectively) in the constraint graphs. Also note that each edge in $G_V(F)$ ($G_V(F)$, respectively) corresponds to a block in the floorplan.

Definition 4. Two mosaic floorplans are *equivalent mosaic floorplans* if and only if they have identical horizontal constraint graphs and vertical constraint graphs.

Thus all three floorplans shown in Fig. 1 (a), (b), and (c) are equivalent mosaic floorplans. Note that the floorplan in Fig. 1(c) is obtained from the floorplan in Fig. 1(b) by sliding the horizontal line segment between the blocks g and d downward; the horizontal line segment between the blocks f and c upward; the vertical line segment between the blocks a and b to the right.

1.2. Applications of floorplans and mosaic floorplans

Floorplans and mosaic floorplans are used in the first major stage (called floorplanning) in the physical design cycle of VLSI (Very Large Scale Integration) circuits [10]. The blocks in a floorplan correspond to the components of a VLSI chip. The floorplanning stage is used to plan the relative position of the circuit components. At this stage, the blocks do not have specific sizes assigned to them yet. So only the position relationship between the blocks are considered.

For a floorplan, the wires between two blocks run cross their common boundary. In this setting, two equivalent floorplans provide the same connectivity between blocks. For a mosaic floorplan, the line segments are the wires. Any block with a line segment on its boundary can be connected to the wires represented by the line segment. In this setting, two equivalent mosaic floorplans provide the same connectivity between blocks.

One of the main problems in this area is to find a short binary representation of floorplans and mosaic floorplans. These representations are used by various algorithms to generate floorplans in order to solve various VLSI layout optimization problems.

1.3. Baxter permutations

Baxter permutations are a set of permutations defined by prohibited subsequences. They were first introduced in [3]. [1] formally defines Baxter permutations. A Baxter permutation on $\{1, 2, \dots, n\}$ is a permutation $\pi = (\sigma_1, \sigma_2, \dots, \sigma_n)$ such that there are no four indices $1 \leqslant i < j < k < l \leqslant n$ such that

1. $\sigma_k < \sigma_i = \sigma_l < \sigma_j$; or
2. $\sigma_j < \sigma_l + 1 = \sigma_i < \sigma_k$.

It was shown in [8] that the set of Baxter permutations has one-to-one correspondences to many interesting objects in the so-called *Baxter combinatorial family*. For examples, [4] showed that *plane bipolar orientations* with n edges have a one-to-one correspondence with Baxter permutations of length n. [5] establishes a relationship between Baxter permutations and pairs of alternating sign matrices.

In particular, it was shown in [1,6,18] that mosaic floorplans are one of the objects in the Baxter combinatorial family. A simple and efficient one-to-one correspondence between mosaic floorplans and Baxter permutations was established in [1,6]. As a result, any binary representation of mosaic floorplans can also be converted to a binary representation of Baxter permutations.

1.4. Previous work on representations of floorplans and mosaic floorplans

Because of their applications in VLSI physical design, the representations of floorplans and mosaic floorplans have been studied extensively by mathematicians, computer scientists and electrical engineers. Although their definitions are similar, the combinatorial properties of floorplans and mosaic floorplans are quite different. The following is a partial list of previous research on floorplans and mosaic floorplans.

Floorplans There is no known formula for calculating $F(n)$, the number of n-block floorplans. The first few values of $F(n)$ (starting from $n = 1$) are $\{1, 2, 6, 24, 116, 642, 3938, \dots\}$. Researchers have been trying to bound the range of $F(n)$. In [2], it was shown that there exists a constant $c = \lim_{n\to\infty}(F(n))^{1/n}$ and $11.56 < c < 28.3$. This means that $11.56^n \leqslant F(n) \leqslant 28.3^n$ for large n. The upper bound of $F(n)$ is reduced to $F(n) \leqslant 13.5^n$ in [7].

Algorithms for generating floorplans were presented in [12]. In [16], a $(5n - 5)$-bit binary string representation of n-block floorplans was found. A different $5n$-bit binary string representation of n-block floorplans was presented in [17]. The shortest known binary string representation of n-block floorplans was given in [15]. This representation uses $(4n - 4)$ bits.

Since $F(n) \geqslant 11.56^n$ for large n [2], any binary string representation of n-block floorplans must use at least $\log_2 11.56^n = n \log_2 11.56 \approx 3.531n$ bits. Closing the gap between the known $(4n - 4)$-bit binary representation and the $3.531n$ lower bound remains an open research problem [15].

Mosaic floorplans It was shown in [6] that the set of n-block mosaic floorplans has a one-to-one correspondence to the set of Baxter permutations, and the number of n-block mosaic floorplans equals the nth *Baxter number* $B(n)$, which is defined as follows:

$$B(n) = \binom{n+1}{1}^{-1} \binom{n+1}{2}^{-1} \sum_{r=0}^{n-1} \binom{n+1}{r} \binom{n+1}{r+1} \binom{n+1}{r+2}$$

In [14], it was shown that $B(n) = \Theta(8^n/n^4)$. The first few Baxter numbers (starting from $n = 1$) are $\{1, 2, 6, 22, 92, 422, 2074, \dots\}$.

There is a long list of papers on representation problem of mosaic floorplans.

[11] proposed a *sequence pair* (SP) representation. Two sets of permutations are used to represent the position relations between blocks. The length of the representation is $2n\log_2 n$ bits.

[9] proposed a *corner block list* (CB) representation for mosaic floorplans. The representation consists of a list S of blocks, a binary string L of $(n-1)$ bits, and a binary string T of $2n-3$ bits. The total length of the representation is $(3n+n\log_2 n)$ bits.

[19] proposed a *twin binary sequences* (TBS) representation for mosaic floorplans. The representation consists of 4 binary strings $(\pi, \alpha, \beta, \beta')$, where π is a permutation of integers $\{1, 2, \dots, n\}$, and the other three strings are n or $(n-1)$ bits long. The total length of the representation is $3n+n\log_2 n$.

A common feature of the above representations is that each block in the mosaic floorplan is given an explicit name (such as an integer between 1 and n). They all use at least one list (or permutation) of these names in the representation. Because at least $\log_2 n$ bits are needed to represent every integer in the range $[1, n]$, the length of these representations is inevitably at least $n\log_2 n$ bits.

A different approach was introduced in [18]. They use a pair of *twin pair binary trees* t_1 and t_2 to represent mosaic floorplans. The blocks of the mosaic floorplan are not given explicit names. Rather, the shape of the two trees t_1 and t_2 are used to encode the position relations of blocks. In this representation, each tree consists of $2n$ nodes. Thus, each tree can be encoded by using $4n$ bits. So the total length of the representation is $8n$ bits. They also proposed an alternate representation using a pair of n-node trees. However, the nodes in the two trees are given names, and the length of the representation is at least $2n\log_2 n$.

In [13], a representation called *quarter-state-sequence* (QSS) was presented. It uses a Q sequence that represents the configuration of one of the corners of the mosaic floorplan. The length of the Q sequence representation is $4n$ bits. This is the best known representation for mosaic floorplans.

Because the number of n-block mosaic floorplans equals the nth Baxter number, at least $\log_2 B(n) = \log_2 \Theta(8^n/n^4) = 3n - o(n)$ bits are needed to represent mosaic floorplans.

1.5. Our main result

Theorem 1. *The set of n-block mosaic floorplans and the set of Baxter permutations of length n can be represented by $(3n-3)$ bits, which is optimal up to an additive lower order term.*

Most binary representations of mosaic floorplans discussed in Section 1.4 are fairly complex. In contrast, the representation introduced in this paper is very simple and easy to implement.

By using the simple one-to-one correspondence between mosaic floorplans and Baxter permutations described in [1], the methods presented in this paper also work on Baxter permutations. Hence, the optimal representation of mosaic floorplans also leads to an optimal representation of Baxter permutations and all other objects in the Baxter combinatorial family.

In this paper, these algorithms for encoding and decoding mosaic floorplans and Baxter permutations are presented. The algorithms have the same asymptotic runtime and space.

2. Optimal binary representation of mosaic floorplans

In this section, we describe our optimal representation of mosaic floorplans.

2.1. Standard form of mosaic floorplans

In the following, we introduce the notion of *standard form* of mosaic floorplans, which plays a central role in our representation.

Let M be a mosaic floorplan. Let h be a horizontal line segment in M. The *upper segment set* of h and the *lower segment set* of h are defined as the following:

ABOVE(h) = the set of vertical line segments of M that intersect h and are above h.
BELOW(h) = the set of vertical line segments of M that intersect h and are below h.

Similarly, for a vertical line segment v in M, the *left segment set* of v and the *right segment set* of h are defined as the following:

LEFT(v) = the set of horizontal segments of M that intersect v and are on the left of v.
RIGHT(v) = the set of horizontal segments of M that intersect v and are on the right of v.

Definition 5. A mosaic floorplan M is in *standard form* if the following hold:

1. For every horizontal segment h in M, all vertical segments in ABOVE(h) appear to the right of all vertical segments in BELOW(h). (See Fig. 3(a).)

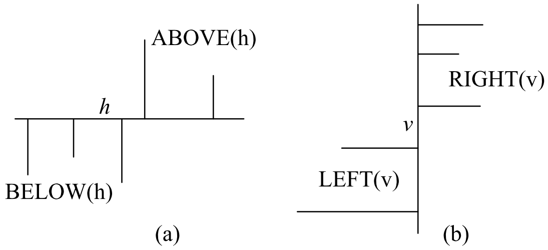

Fig. 3. Standard form of mosaic floorplans.

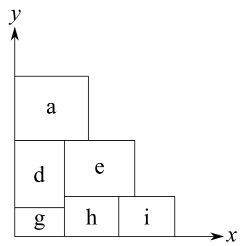

Fig. 4. A staircase with $n = 6$ blocks and $m = 3$ steps that is obtained from the mosaic floorplan in Fig. 1(c) by deleting the blocks b, c and f.

2. For every vertical segment v in M, all horizontal segments in RIGHT(v) appear above all horizontal segments in LEFT(v). (See Fig. 3(b).)

The mosaic floorplan shown in Fig. 1(c) is the standard form of mosaic floorplans shown in Fig. 1 (a) and (b).

The standard form M_{standard} of a mosaic floorplan M can be obtained by sliding its vertical and horizontal line segments. Because of the equivalence definition of mosaic floorplans, M_{standard} and M are considered the same mosaic floorplans. For a given M, M_{standard} can be obtained in linear time by using the horizontal constraint graphs and vertical constraint graphs described in [9]. From now on, all mosaic floorplans are assumed to be in standard form.

2.2. Staircases

Definition 6. A *staircase* is an object that satisfies the following conditions:

1. The border contains a line segment on the x-axis and a line segment on the y-axis.
2. The remainder of the border consists of vertical and horizontal line segments that only form ⌝ and ⌞ junctions.
3. The interior is divided into smaller rectangular subsections by horizontal and vertical line segments.
4. No four subsections meet at the same point.

A *step* of a staircase S is a horizontal line segment on the border of S, excluding the x-axis. Fig. 4 shows a staircase with $n = 6$ blocks and $m = 3$ steps. Note that a mosaic floorplan is just a special case of a staircase with $m = 1$ step.

2.3. Deletable rectangles

Definition 7. A *deletable rectangle* of a staircase S is a block that satisfies the following conditions:

1. Its top edge is completely contained in the border of S.
2. Its right edge is completely contained in the border of S.

In the staircase shown in Fig. 4, the block a is the only deletable rectangle. The concept of deletable rectangles is a key idea for the methods introduced in this paper. This concept was originally defined in [15] for their $(4n - 4)$-bit representation of floorplans. However, a modified definition of deletable rectangles is used in this paper to create a $(3n - 3)$-bit representation of mosaic floorplans.

Lemma 1. *The removal of a deletable rectangle from a staircase results in another staircase unless the original staircase contains only one block.*

Proof. Let S be a staircase with more than one block and let r be a deletable rectangle in S. Define S' to be the object that results when r is removed from S. Because the removal of r still leaves S' with at least one block, the border of S'

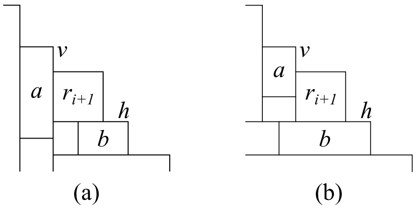

Fig. 5. Proof of Lemma 2.

still contains a line segment on the x-axis and a line segment on the y-axis, so condition (1) of a staircase holds for S'. Removing r will not cause the remainder of the border to have an increasing line segment because the right edge of r must be completely contained in the border, so condition (2) of a staircase also holds for S'. The removal of r does not form new line segments, so the interior of S' will still be divided into smaller rectangular subsections by vertical and horizontal line segments, and no four subsections in S' will meet at the same point. Thus, conditions (3) and (4) of a staircase hold for S'. Therefore, S' is a staircase. □

We can now outline the basic ideas of our representation. Given a mosaic floorplan M, we remove deletable rectangles of M one-by-one. By Lemma 1, this results in a sequence of staircases, until only one block remains. We record necessary location information of these deletable rectangles (which will be the binary representation of M) so that we can reconstruct the original floorplan M. However, if there are multiple deletable rectangles for these staircases, we will have to use more bits than we can afford. Fortunately, the following key lemma shows that this does not happen.

Lemma 2. *Let M be an n-block mosaic floorplan in standard form. Let $S_n = M$, and let S_{i-1} $(2 \leqslant i \leqslant n)$ be the staircase obtained by removing a deletable rectangle r_i from S_i.*

1. *There is a single, unique deletable rectangle in S_i for $1 \leqslant i \leqslant n$.*
2. *r_{i-1} is adjacent to r_i for $2 \leqslant i \leqslant n$.*

Proof. The proof is by reverse induction.

Clearly, $S_n = M$ has only one deletable rectangle located at the top right corner of M.

Assume that S_{i+1} $(i \leqslant n-1)$ has exactly one deletable rectangle r_{i+1}. Let h be the horizontal line segment in S_{i+1} that contains the bottom edge of r_{i+1}, and let v be the vertical line segment in S_{i+1} that contains the left edge of r_{i+1} (see Fig. 5). Let a be the uppermost block in S_{i+1} whose right edge aligns with v, and let b be the rightmost block in S_{i+1} whose top edge aligns with h. After r_{i+1} is removed from S_{i+1}, a and b are the only candidates for deletable rectangles of the resulting staircase S_i. There are two cases:

1. The line segments h and v form a $\vdash$-junction (see Fig. 5(a)). Then, the bottom edge of a must be below h because M is a standard mosaic floorplan, and a is not a deletable rectangle in S_i. Thus, the block b is the only deletable rectangle in S_i.
2. The line segments h and v form a $\perp$-junction (see Fig. 5(b)). Then, the left edge of b must be to the left of v because M is a standard mosaic floorplan, and b is not a deletable rectangle in S_i. Thus, the block a is the only deletable rectangle in S_i.

In both cases, only one deletable rectangle r_i (which is either a or b) is revealed when the deletable rectangle r_{i+1} is removed. Because there is only one deletable rectangle in $S_n = M$, all subsequent staircases contain exactly one deletable rectangle. Thus, (1) is true. Also, r_{i+1} is adjacent to r_i in both cases, so (2) is true. □

Let S be a staircase and r be a deletable rectangle of S whose top side is on the k-th step of S. There are four types of deletable rectangles.

1. Type $(0, 0)$:
   (a) The top side of r is the entire k-th step.
   (b) The right side of r intersects the $(k-1)$-st step.
   (c) The deletion of r decreases the number of steps by one.
2. Type $(0, 1)$:
   (a) The top side of r is only a part of the k-th step.
   (b) The right side of r intersects the $(k-1)$-st step.
   (c) The deletion of r does not change the number of steps.

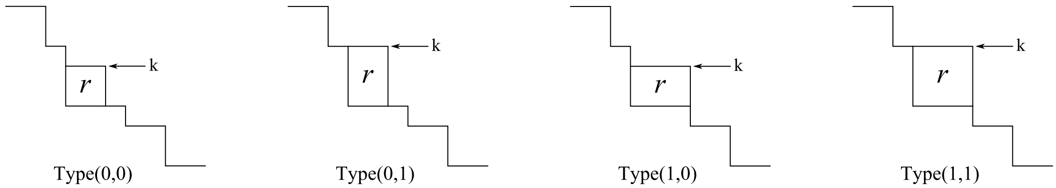

Fig. 6. The four types of deletable rectangles.

3. Type $(1, 0)$:
   (a) The top side of r is the entire k-th step.
   (b) The right side of r is only a part of the right side of the k-th step (namely the right-bottom corner of r is a $\dashv$ shape junction).
   (c) The deletion of r does not change the number of steps.
4. Type $(1, 1)$:
   (a) The top side of r is only a part of the k-th step.
   (b) The right side of r is only a part of the right side of the k-th step (namely the right-bottom corner of r is a $\dashv$ shape junction).
   (c) The deletion of r increases the number of steps by one.

2.4. Optimal binary representation

Our binary representation of mosaic floorplans depends on the fact that a mosaic floorplan M is a special case of a staircase and the fact that the removal of a deletable rectangle from a staircase results in another staircase. The binary string used to represent M records the unique sequence of deletable rectangles that are removed in this process. The information stored by this binary string enable us to reconstruct the original mosaic floorplan M.

A 3-bit binary string is used to record the information for each deletable rectangle r_i. The string has two parts: The type and the location of r_i. To record the type of r_i, the bits corresponding to its type is stored directly. To store the location, we note that, by Lemma 2, two consecutive deletable rectangles r_i and r_{i-1} are adjacent to each other. Thus, they must share either a horizontal edge or a vertical edge. A single bit can be used to record the location of r_i with respect to r_{i-1}: a 1 if they share a horizontal edge, and a 0 if they share a vertical edge.

Encoding procedure. Let M be the n-block mosaic floorplan to be encoded. Starting from $S_n = M$, remove the unique deletable rectangles r_i, where $2 \leqslant i \leqslant n$, one-by-one. For each deletable rectangle r_i, two bits are used to record the type of r_i, and one bit is used to record the type of the common boundary shared by r_i and r_{i-1}.

Decoding procedure. The decoding procedure simply reverses the process of removing deletable rectangles. The process starts with the staircase S_1, which is a single rectangle. Each staircase S_{i+1} can be reconstructed from the staircase S_i by using the three-bit code for the deletable rectangle r_{i+1}. The three-bit code records the type of r_{i+1} and the type of edge shared by r_i and r_{i+1}, so r_{i+1} can be uniquely added to S_i. Thus, the decoding procedure can reconstruct the original mosaic floorplan $S_n = M$.

Fig. 7 shows an example of the reconstruction of a mosaic floorplan from its representation:

000 011 101 000 110 111

The lower left block of the mosaic floorplan M (which is the only block of S_1) does not need any information to be recorded. Each of the other blocks of M needs three bits. Thus the total length of the binary representation of M is $(3n - 3)$ bits. This completes the proof of Theorem 1.

3. Optimal binary representation of Baxter permutations

Two procedures were presented in [1]: **FP2BP** is a bijection from mosaic floorplans to Baxter permutations; **BP2FP** is a bijection from Baxter permutations to mosaic floorplans. By combining these two procedures and the encoding and decoding procedures for mosaic floorplans presented in this paper, an optimal binary representation of Baxter permutations is obtained. While the procedure **FP2BP** is straightforward, the procedure **BP2FP** is more complicated. In this section, an encoding procedure that maps a Baxter permutation π directly to a $(3n - 3)$-bit binary string is presented, bypassing the intermediate mosaic floorplan.

For completeness, the procedures **FP2BP** and **BP2FP** are presented below. Given an n-block mosaic floorplan M, a mosaic floorplan of $n - 1$ blocks can be obtained by using *block deletion* operation introduced in [9]:

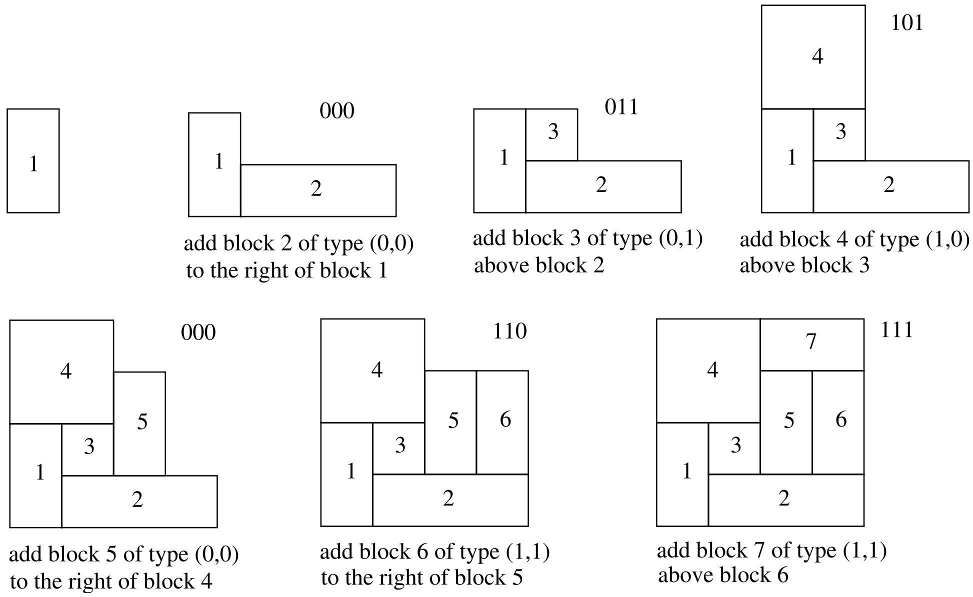

Fig. 7. An example of the presentation.

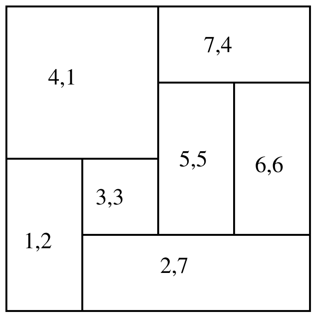

Fig. 8. An example of **FP2BP** procedure.

Definition 8. Let M be a mosaic floorplan with $n > 1$ blocks and let b be the top-left block in M. If the bottom-right corner of b is a $\dashv$ ($\perp$) junction, then one can delete b from M by shifting its bottom (right) edge upwards (leftwards), while pulling the T-junctions attached to it until the edge hits the top (left) boundary of M.

Note that we can delete a block from the lower-left corner of M in a similar manner.

Algorithm FP2BP
Input: A mosaic floorplan M with n blocks.
Output: A Baxter permutation on $\{1, \dots, n\}$.
1. Label the blocks of M according to their deletion order from the top-left corner.
2. Return the permutation of the labels obtained by deleting the blocks of M from the bottom-left corner.

Fig. 8 shows an example of **FP2BP** procedure. Each block in the figure has two numbers associated with it: The first number is the order obtained in step 2 of **FP2BP**. The second number is the name of the block obtained in step 1 of **FP2BP** procedure. After performing step 2, the procedure outputs the permutation $\pi = 2731564$.

Algorithm BP2FP
Input: A Baxter permutation $\pi = (\sigma_1, \sigma_2, \dots, \sigma_n)$.
Output: A mosaic floorplan M with n blocks.
1. Draw the block 1, label it by σ_1.
2. Construct an $n \times n$ grid within the block 1.
3. **for** $i = 2$ to n **do**:

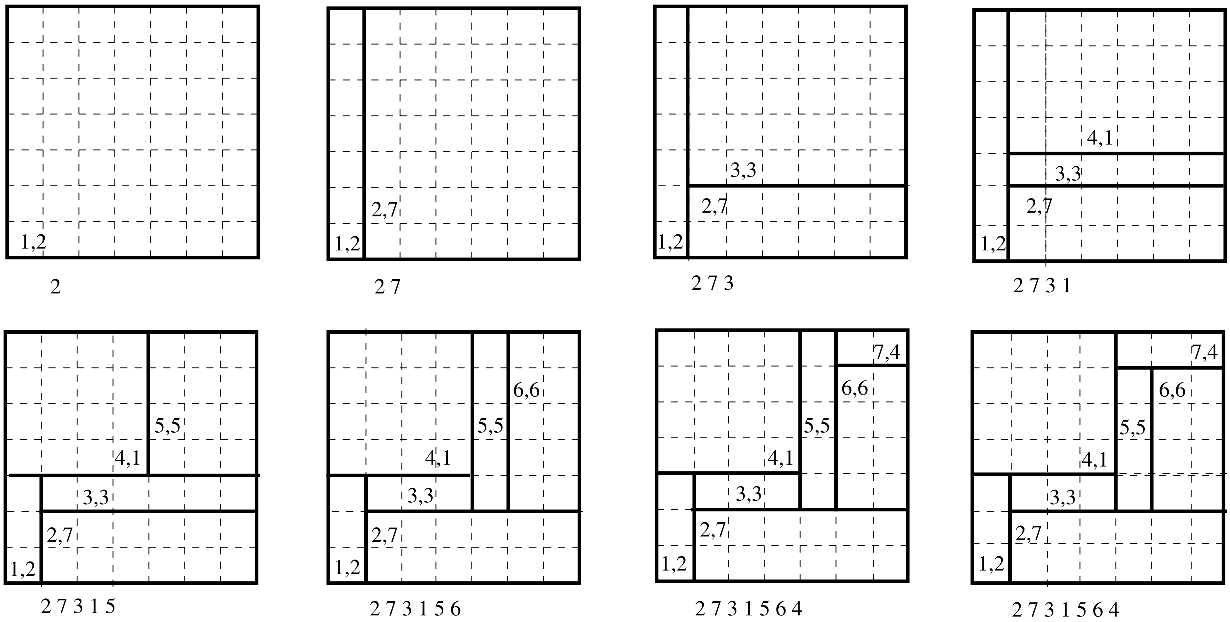

Fig. 9. An example of **BP2FP** procedure.

if $\sigma_i < \sigma_{i-1}$ **then**
    Slice the top-right block by a horizontal segment at the i-th level of the grid;
    Name the new top-right block the block i, and label it by σ_i;
    while the label σ' of the block b to the left of the block i is larger than σ_i **do**
        Extend the block i leftwards (at the expense of the block b);
    end while;
else
    Slice the top-right block by a vertical segment at the i-th level of the grid;
    Name the new top-right block the block i, and label it by σ_i;
    while the label σ' of the block b below the block i is smaller than σ_i **do**
        Extend the block i downwards (at the expense of the block b);
    end while;
end if
end for

Fig. 9 shows an example of **BP2FP** procedure. It shows, step-by-step, the mosaic floorplan generated from the permutation $\pi = 2731564$. Note that after the block labeled by 5 is added, it is extended downwards (at the expense of the block labeled by 3). Similarly, after the block labeled by 4 is added, it is extended leftwards (at the expense of the block labeled by 5).

Definition 9. Let $\pi = (\sigma_1, \ldots, \sigma_n)$ be a permutation. For each index i $(1 \leqslant i \leqslant n)$, define:

1. The *right larger entry of* i, denoted by $rl(i)$, is the first σ_j such that $j > i$ and $\sigma_j > \sigma_i$. (If no such entry exists, define $rl(i) = n + 1$.)
2. The *left larger entry of* i, denoted by $ll(i)$, is the last σ_j such that $j < i$ and $\sigma_j > \sigma_i$. (If no such entry exists, define $rl(i) = n + 1$.)
3. The *right smaller entry of* i, denoted by $rs(i)$, is the first σ_j such that $j > i$ and $\sigma_j < \sigma_i$. (If no such entry exists, define $rs(i) = 0$.)
4. The *left smaller entry of* i, denoted by $ls(i)$, is the last σ_j such that $j < i$ and $\sigma_j < \sigma_i$. (If no such entry exists, define $ls(i) = 0$.)

For the permutation $\pi = (2731564)$, we have $ll(3) = 7$, $rl(3) = 5$, $ls(3) = 2$ and $rs(3) = 1$. The following properties of Algorithm **BP2FP** were proved in [1]:

Property 1. *Let M be the mosaic floorplan generated by Algorithm* **BP2FP** *from a Baxter permutation $\pi = (\sigma_1, \ldots, \sigma_n)$. For each index i $(1 \leqslant i \leqslant n)$:*

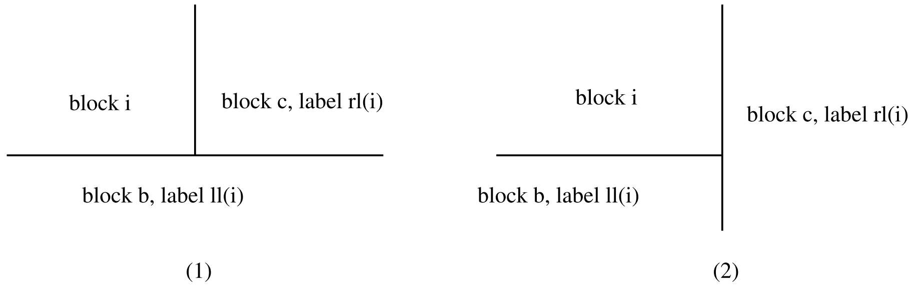

Fig. 10. The proof of Lemma 3.

1. *If $\sigma_{i-1} < \sigma_i$, then the block $i-1$ is to the left of the block i and they share a vertical boundary. If $\sigma_{i-1} > \sigma_i$, then the block $i-1$ is below the block i and they share a horizontal boundary.*
2. *The block b with label $ll(i)$ is below the block i and they share a horizontal boundary. (If $ll(i) = n+1$, then the block i is on the bottom boundary of M.) The block c with label $rl(i)$ is to the right of the block i and they share a vertical boundary. (If $rl(i) = n+1$, then the block i is on the right boundary of M.)*
3. *The block d with label $ls(i)$ is to the left of the block i and they share a vertical boundary. (If $ls(i) = 0$, then the block i is on the left boundary of M.) The block e with label $rl(i)$ is above the block i and they share a horizontal boundary. (If $rs(i) = 0$, then the block i is on the top boundary of M.)*

Let M be the mosaic floorplan generated by **BP2FP** from a Baxter permutation $\pi = (\sigma_1, \dots, \sigma_n)$. Let C_M be the $(3n-3)$-bit code for M generated by the encoding procedure in Section 2. We will generate C_M directly from π without using the intermediate floorplan M. Recall that C_M contains a 3-bit code word code_i for each block i $(2 \leqslant i \leqslant n)$ of M:

- $\text{code}_i[1]$ indicates the type of the T-junction of the lower-right corner of the block i:

$$\text{code}_i[1] = \begin{cases} 0 & \text{if the lower-right corner of the block } i \text{ is a } \perp \text{ junction} \\ 1 & \text{if the lower-right corner of the block } i \text{ is a } \dashv \text{ junction} \end{cases}$$

- $\text{code}_i[2]$ indicates the type of the T-junction of the upper-left corner of the block i:

$$\text{code}_i[2] = \begin{cases} 0 & \text{if the upper-left corner of the block } i \text{ is a } \vdash \text{ junction} \\ 1 & \text{if the upper-left corner of the block } i \text{ is a } \top \text{ junction} \end{cases}$$

- $\text{code}_i[3]$ indicates the type of the common boundary of the block i and the block $i-1$:

$$\text{code}_i[3] = \begin{cases} 0 & \text{if the common boundary of the block } i \text{ and the block } i-1 \text{ is vertical} \\ 1 & \text{if the common boundary of the block } i \text{ and the block } i-1 \text{ is horizontal} \end{cases}$$

Lemma 3. *For each i $(2 \leqslant i \leqslant n)$, code_i for the block i can be determined by the following rules*:

1. *If $\sigma_i > \sigma_{i-1}$, then $\text{code}_i[3] = 0$. If $\sigma_i < \sigma_{i-1}$, then $\text{code}_i[3] = 1$.*
2. *If $rl(i) \leqslant ll(i)$, then $\text{code}_i[1] = 0$. If $rl(i) > ll(i)$, then $\text{code}_i[1] = 1$.*
3. *If $rs(i) \geqslant ls(i)$, then $\text{code}_i[2] = 0$. If $rs(i) < ls(i)$, then $\text{code}_i[2] = 1$.*

Proof. 1. Immediate from Statement 1 in Property 1.

2. Let b be the block with the label $ll(i)$ and c be the block with the label $rl(i)$. By Statement 2 in Property 1, the block b is below the block i and they share a horizontal boundary, and the block c is to the right of the block i and they share a vertical boundary. By the definition of $rl(i)$ and $rl(i)$, we have: for any index k with $b < k < c$, $ll(i) > \sigma_k$ and $rl(i) > \sigma_k$. There are two cases:

- $ll(i) \geqslant rl(i)$. First assume $ll(i) > rl(i)$. Then $ll(c) = ll(i)$. By Statement 2 in Property 1, the block b is below the block c and they share a horizontal boundary. So the lower-right corner of the block i is a $\perp$ junction and $\text{code}_i[1] = 0$. (See Fig. 10(1).)
  The condition $ll(i) = rl(i)$ can occur only when $\sigma_i = n$ (in this case $ll(i) = rl(i) = n+1$). In this case the block i must be located at the lower-right corner of M, and the $\text{code}_i[1] = 0$ by default.
- $ll(i) < rl(i)$. Then $rl(b) = rl(i)$. By Statement 2 in Property 1, the block c is to the right of the block b and they share a vertical boundary. So the lower-right corner of the block i is a $\dashv$ junction and $\text{code}_i[1] = 1$. (See Fig. 10(2).)

3. The proof is similar to the proof of the Statement 2. $\square$

By using the rules in Lemma 3, we can easily generate a $(3n-3)$-bit binary code from a Baxter permutation π.

4. Conclusion

In this paper, we introduced a binary representation of n-block Mosaic floorplans. The representation uses $(3n-3)$ bits. Since any representation of n-block mosaic floorplans requires at least $(3n-o(n))$ bits [14], our representation is optimal (up to an additive lower term). Our representation is very simple and easy to implement.

Mosaic floorplans are known to have a simple one-to-one correspondence with Baxter permutations. So the method used to represent mosaic floorplans in this paper also lead to an optimal $(3n-3)$ bits representation of Baxter permutation of length n, and all objects in the Baxter combinatorial family.

References

[1] Eyal Ackerman, Gill Barequet, Ron Y. Pinter, A bijection between permutations and floorplans, and its applications, Discrete Appl. Math. 154 (2006) 1674–1684.
[2] Kazuyuki Amano, Shin'ichi Nakano, Katsuhisa Yamanaka, On the number of rectangular drawings: Exact counting and lower and upper bounds, IPSJ SIG Notes 2007-AL-115-5C (2007) 33–40.
[3] G. Baxter, On fixed points of the composite of commuting functions, Proc. Amer. Math. Soc. 15 (1964) 851–855.
[4] Nicolas Bonichon, Mireille Bousquet-Mélou, Éric Fusy, Baxter permutations and plane bipolar orientations, Electron. Notes Discrete Math. 31 (2008) 69–74.
[5] Hal Canary, Aztec diamonds and Baxter permutations, Electron. J. Comb. 17 (2010), 12 pp.
[6] S. Dulucq, O. Guibert, Baxter permutations, Discrete Math. 180 (1998) 143–156.
[7] Ryo Fujimaki, Youhei Inoue, Toshihiko Takahashi, An asymptotic estimate of the numbers of rectangular drawings or floorplans, in: Proceedings 2009 IEEE International Symposium on Circuits and Systems, 2009, pp. 856–859.
[8] Samuele Giraudo, Algebraic and combinatorial structures on Baxter permutations, in: DMTCS Proceedings, 23rd International Conference on Formal Power Series and Algebraic Combinatorics (FPSAC 2011), 2011.
[9] Xianlong Hong, Gang Huang, Yici Cai, Jiangchun Gu, Sheqin Dong, Chung-Kuan Cheng, Jun Gu, Corner-block list: An effective and efficient topological representation of non-slicing floorplan, in: Proceedings of the International Conference on Computer Aided Design (ICCAD'00), 2000, pp. 8–12.
[10] Thomas Lengauer, Combinatorial Algorithms for Integrated Circuit Layout, John Wiley & Sons, 1990.
[11] Hiroshi Murata, Kunihiro Fujiyoshi, Rectangle-packing-based module placement, in: Proceedings of the International Conference on Computer Aided Design (ICCAD'95), 1995, pp. 472–479.
[12] Shin'ichi Nakano, Enumerating floorplans with n rooms, in: Proceedings 12th International Symposium on Algorithms and Computation (ISAAC'01), in: Lecture Notes in Computer Science, vol. 2223, 2001, pp. 107–115.
[13] Keishi Sakanushi, Yoji Kajitani, Dinesh P. Mehta, The quarter-state-sequence floorplan representation, IEEE Trans. Circ. Syst. I, Fund. Theor. Appl. 50 (3) (2003) 376–386.
[14] Zion Cien Shen, Chris C.N. Chu, Bounds on the number of slicing, mosaic, and general floorplans, IEEE Trans. Comput.-Aided Des. Integr. Circ. Syst. 22 (10) (2003) 1354–1361.
[15] Toshihiko Takahashi, Ryo Fujimaki, Youhei Inoue, A $(4n-4)$-bit representation of a rectangular drawing or floorplan, in: Proceedings 15th International Computing and Combinatorics Conference (COCOON'09), in: Lecture Notes in Computer Science, vol. 5609, 2009, pp. 47–55.
[16] Katsuhisa Yamanaka, Shin'ichi Nakano, Coding floorplans with fewer bits, IEICE Trans. Fund. E 89 (5) (2006) 1181–1185.
[17] Katsuhisa Yamanaka, Shin'ichi Nakano, A compact encoding of rectangular drawings with efficient query supports, in: Proceedings of the 3rd International Conference on Algorithmic Aspects in Information and Management (AAIM'07), in: Lecture Notes in Computer Science, vol. 4508, 2007, pp. 68–81.
[18] Bo Yao, Hongyu Chen, Chung-Kuan Cheng, Ronald Graham, Floorplan representation: Complexity and connections, ACM Trans. Des. Automat. Electron. Syst. 8 (1) (2003) 55–80.
[19] E.E.Y. Young, C.C.N. Chu, Z.C. Shen, Twin binary sequences: A nonredundant representation for general nonslicing floorplan, IEEE Trans. Comput.-Aided Des. Integr. Circ. Syst. 22 (4) (2003) 457–469.