Journal of Information Processing Vol.22 No.4 634–637 (Oct. 2014)

[DOI: 10.2197/ipsjjip.22.634]

Recommended Paper

A Compact Code for Rectangular Drawings with Degree Four Vertices

Toshihiko Takahashi[1,a)]

Received: December 28, 2013, Accepted: April 4, 2014

Abstract: A subdivision of a rectangle into rectangular faces with horizontal and vertical line segments is called a rectangular drawing or floorplan. Several encodings of rectangular drawings have been published; however, most of them deal with rectangular drawings without vertices of degree four. Recently, Saito and Nakano developed two compact encodings for general rectangular drawings, that is, which allows vertices of degree four. The two encodings respectively need $6f - 2n_4 + 6$ bits and $5f - 5$ bits for rectangular drawings with f inner faces and n_4 degree four vertices. The best encoding of the two depends on the number of vertices of degree four, that is, the former is the better if $2n_4 > f + 11$; otherwise the latter is the better. In this paper, we propose a new encoding of general rectangular drawings with $5f - n_4 - 6$ bits for $f \geq 2$, which is the most compact regardless of n_4.

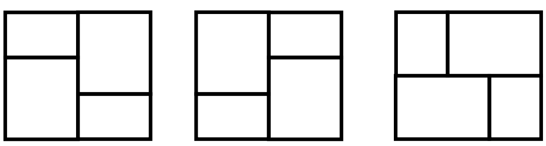
Keywords: rectangular drawing, floorplan, plane graph, encoding

1. Introduction

A *rectangular drawing* or *floorplan* is a subdivision of a rectangle with horizontal and vertical line segments. Usually no two line segments are allowed to decussate, that is, an ordinary rectangular drawing has no crisscross intersections of line segments (**Fig. 1** (a)–(c)). Two rectangular drawings are equivalent if (i) they have the same adjacent relations between the subdividing line segments and the rectangles and (ii) they have the same adjacent relations between the rectangles. We consider the direction of rectangular drawing. Thus, the three rectangular drawings in Fig. 1 are all different.

Subdivisions of rectangles are also called rectangular partitions or mosaic floorplans. However, two rectangular partitions or mosaic floorplans are equivalent if only condition (i) is satisfied, that is condition (ii) is ignored. See Refs. [9], [10], [11], [12] for encodings. A survey of these encodings is also available [1].

For application in VLSI physical design, several encodings of rectangular drawings have been published: For example, H-Sequence [2], EQ-Sequence [3], FT-Squeeze [5], and so on. The bit length of codes is a mesure of encoding schemes [4]. Takahashi, Fujimaki, and Inoue have given a $(4f - 4)$-bit encoding of an ordinary rectangular drawing, where f is the number of rectangles (inner faces) of a rectangular drawing [7].

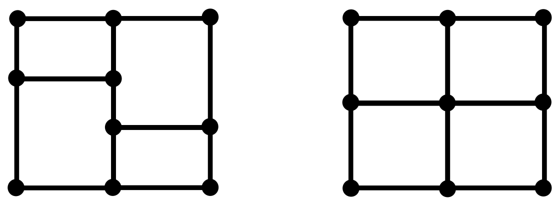


(a) (b) (c)

Fig. 1 Three different ordinary rectangular drawings.

Rectangular drawings can be seen as special planar drawings of graphs: The vertices are the intersections of line segments and the edges are line segments between the vertices (**Fig. 2** (a)). From the viewpoint of graph drawing, encodings of rectangular drawings with vertices of degree four are strongly desired (Fig. 2 (b)). In the following, we will consider a rectangular drawing which might have vertices of degree four and call them *general rectangular drawings* (**Fig. 3**). Saito and Nakano developed two compact encodings of general rectangular drawings [8]. The first encoding in Ref. [8] is called code I, which is based on depth-first search of on ordered tree. The bit length of the code I is $6f - 2n_4 + 6$, where n_4 is the number of vertices of degree four. The second one is called code II, which is a pair of the $(4f - 4)$-bit code of ordinary rectangular drawings [7] and information of vertices of degree four. The bit length of the code II is $5f - 5$.

If $2n_4 > f + 11$, code I is the better since $6f - 2n_4 + 6 < 5f - 5$; otherwise code II is. That is, the best encoding of the two depends on the number of vertices of degree four.



(a) (b)

Fig. 2 Rectangular drawings as graphs: (a) without vertices of degree four; (b) with a vertex of degree four.

1 Institute of Natural Science and Technology, Academic Assembly, Niigata University, Niigata 950–2181, Japan

a) takahasi@ie.niigata-u.ac.jp

The initial version of the paper was presented at the FIT2013 workshop held on Sept. 2013. The paper was recommended to be submitted to Journal of Information Processing (JIP) by the chairman of FIT2013.

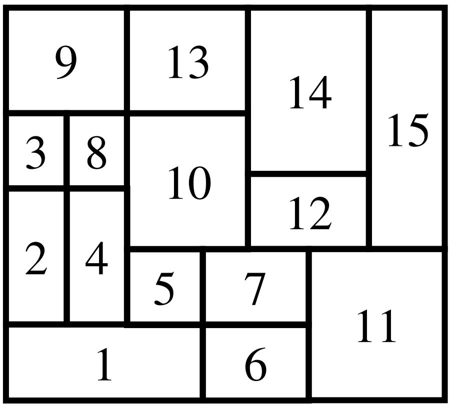


Fig. 3 A general rectangular drawing.

In this paper, we propose a new encoding of general rectangular drawings with $5f - n_4 - 6$ bits for $f \geq 2$, which is the most compact regardless of n_4.

This paper is organized as follows: Section 2 introduces staircase and deletable rectangle, which are variants of those in Ref. [7]. Section 3 gives the encoding and an upper bound of the bit lengths.

2. General Staircase and Deletable Rectangle

Staircase appeared in Ref. [6] for computing the number of rectangular drawings. In this section, a variant is introduced.

2.1 Staircase

Consider a rectangular drawing R placed in the xy-plane so that the bottom-left corner is located at the origin. A *general staircase* for R is a configuration obtained from R by deleting rectangles such that

- the border consists of two line segments on the x-axis and y-axis and a monotonic decreasing rectilinear path i.e., polygonal line of horizontal and vertical line segments, and
- the interior is subdivided into rectangles with horizontal and vertical line segments (**Fig. 4**).

In the following, ‘general’ is omitted for simplicity.

Horizontal line segments of the monotonic decreasing rectilinear path are called *steps*. A rectangle is called a *step rectangle* if its top-right corner is at the right end of a step. For example, the staircase in Fig. 4 has three steps and rectangles 9, 10, and 11 are step rectangles.

The number of inner rectangles of a staircase is also denoted f as in the case of a rectangular drawing. Note that a rectangular drawing is also a staircase with one step.

2.2 Deletable Rectangle

The *deletable rectangle* r of a staircase is the uppermost rectangle among the rectangles satisfying the following four conditions:

(1) The top side of r is wholly contained in the border of the staircase.
(2) The right side of r is wholly contained in the border of the staircase.
(3) The rightward ray from the bottom-right corner of r does not meet a top-left corner of another rectangle.
(4) The upward ray from the top-left corner of r does not meet a bottom right corner of another rectangle except at the initial point of the ray.

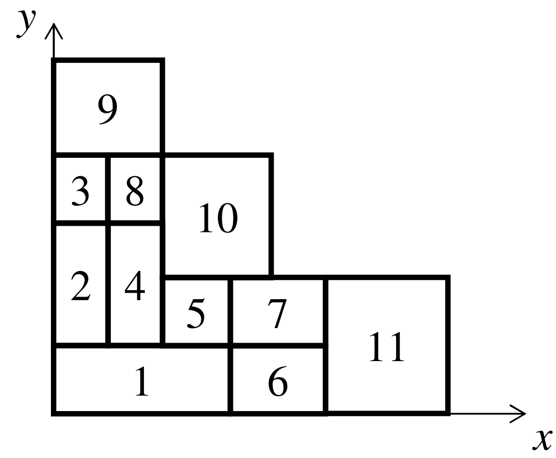


Fig. 4 A general staircase.

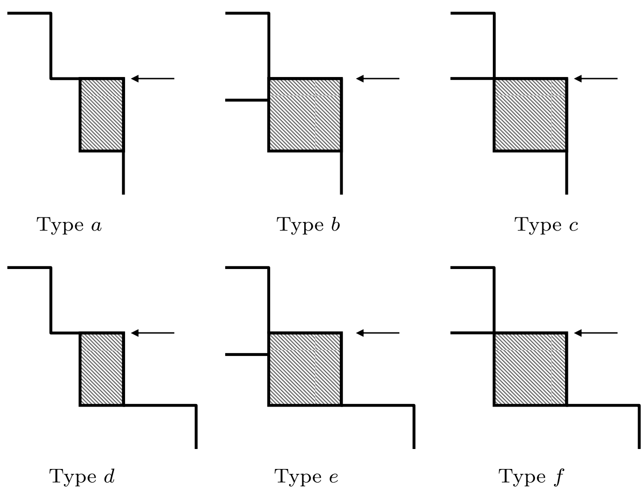


Fig. 5 The six types of deletable rectangles.

Note that the condition 4 has an exception. It is easy to see that the deletable rectangle is uniquely defined for every staircase with $f > 0$: Let the step rectangles be $sr_1, sr_2, \ldots, sr_m$ from the top. The topmost step rectangle sr_1 satisfies the conditions 1 and 4. If sr_1 violates the conditions 2 or 3, sr_2 satisfies the conditions 1 and 4. Similarly, if sr_2 again violates the conditions 2 or 3, sr_3 satisfies the conditions 1 and 4, and so on. On the other hand the bottommost step rectangle sr_m satisfies the conditions 2 and 3.

See the staircase in Fig. 4. Only rectangle 11 satisfies the above four conditions. Rectangle 9 violates condition 3 since the rightward ray from its bottom-right corner meet the top-left corner of rectangle 10. Rectangle 10 also violates condition 3 since the rightward ray from its bottom-right corner meets the top-left corner of rectangle 11. However, rectangle 10 does not violate condition 4: The upward ray from its top-left corner meets the bottom right corner of rectangle 9 at the initial point of the ray, which is a vertex of degree four. Therefore, rectangle 11 is the deletable rectangle in the staircase.

Deletable rectangles are classified into the following six types as shown in **Fig. 5**. Let r be a deletable rectangle of a staircase.

- Group A: the bottom-right corner of r is located at the right end of a step in the resultant staircase, that is, the staircase obtained by deleting r.

- Type a: The top side of r is strictly included in a step. The deletion of r increases the number of the steps of the staircase by one.
- Type b : The top side of r coincides with a step and the degree of the top-left corner of r is three. The deletion of r does not change the number of the steps of the staircase.
- Type c : The top side of r coincides with a step and the degree of the top-left corner of r is four. The deletion of r does not change the number of the steps of the staircase.

• Group B: the bottom-right corner is *not* located at the right end of a step in the resultant staircase.

- Type d: The top side of r is strictly included in a step. The deletion of r does not change the number of the steps of the staircase.
- Type e : The top side of r coincides with a step and the degree of the top-left corner of r is three. The deletion of r decreases the number of the steps of the staircase by one.
- Type f : The top side of r coincides with a step and the degree of the top-left corner of r is four. The deletion of r decreases the number of the steps of the staircase by one.

3. A $(5f - n_4 - 6)$-bit Representation of a General Rectangular Drawing

In this section, we give a variant of the encoding for ordinary rectangular drawings in Ref. [7].

3.1 A String Representation and Encoding

First we give a representation of a rectangular drawing on alphabet $\{0, A, B\}$ as in Ref. [7]. Let S_f and r_f be a rectangular drawing with f rectangles and its deletable rectangle, respectively. The staircase obtained by deleting r_f from S_f has $f - 1$ rectangles. Denote the staircase and its deletable rectangle by S_{f-1} and r_{f-1}, respectively. Again deleting r_{f-1} from S_{f-1}, we obtain staircase S_{f-2} with deletable rectangle r_{f-2}. In this way, we obtain a sequence of staircases $S_f, S_{f-1}, \ldots, S_1$, where S_1 is the staircase with $f = 1$, that is, a single rectangle. Note that the sequence is uniquely determined since the deletable rectangle r_i is unique for S_i $(i = f, \ldots, 2)$.

For the representation, we define the *candidate positions* of staircase S_i $(i = 1, \ldots, f - 1)$. Consider adding rectangle r_{i+1} to staircase S_i and obtaining S_{i+1}. According to the six types of deletable rectangles, the position of the top-left corner of r_{i+1} must be one of the following:

(1) A point on the y-axis above the top step of staircase S_i: In **Fig. 6**, the position indicated by arrow 0.
(2) The right end point of a step of S_i: In Fig. 6, the positions indicated by arrows 1 and 4.
(3) A point on both the right side of a step rectangle and the border of S_i: In Fig. 6, the positions indicated by arrows 2 and 5.
(4) The bottom-right corner of a step rectangle on the border except on the x-axis: In Fig. 6, the positions indicated by arrows 3 and 6.

The above positions whose y-coordinate is equal to or more than that of r_i are called *candidate positions*. Candidate positions are numbered $0, 1, \ldots$ beginning at the top (Fig. 6). Rectangle r_{i+1} must be added to one of the candidate positions of S_i. (In Fig. 6, bold arrows 0,1,2,and 3 indicate the candidate positions. The deletable rectangle is shaded. Thus, for example, position 4 cannot be a candidate: If r_{i+1} were added to position 4, it would not be the deletable rectangle in the resultant staircase S_{i+1}.)

Now we are ready to describe how to reconstruct the sequence of staircases $S_1, S_2, \ldots, S_f$ by consecutively adding rectangles $r_2, r_3, \ldots, r_f$.

First compute the following parameters by consecutively deleting rectangles $r_f, \ldots, r_2$.

- c_i: the candidate position in S_{i-1} at which r_i is added to;
- d_i: the lowest candidate position of S_i.
- T_i: the group of r_i;
- $\delta_i = d_{i-1} - c_i$.

For the example in Fig. 3, the result is shown in **Table 1**.

The location and the type of rectangle r_i are determined by S_{i-1}, c_i, and T_i. Since d_{i-1} is an invariant of S_{i-1}, the location and the type are also determined by S_{i-1}, δ_i, and T_i.

Let string s_i $(i = 2, \ldots, f)$ be the unary representation of δ_i followed by T_i. For the exmaple, $s_2 = 00B, s_3 = 000A, s_4 = B, \ldots, s_{15} = 00A$. The string representation of S_f on alphabet $\{0, A, B\}$ is the concatenation $s_2 s_3 \cdots s_f$. Finally replace A and B in the representation by 10 and 11 to obtain the code, i.e., bit representation of S_f. The code for our example is the following 58-bit code: 0011000101 1010110001 0000000100 0010111100 1100000010 00100010.

Linear time encoding and decoding algorithms are almost the

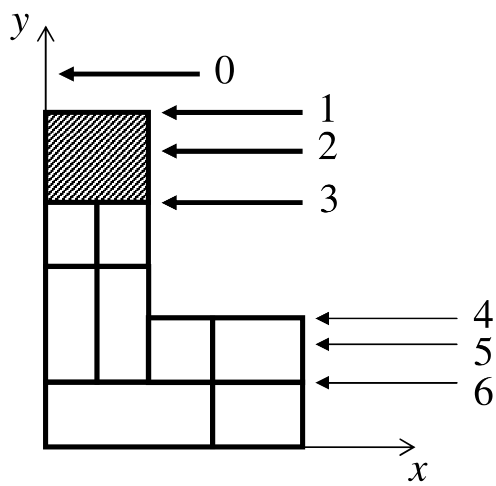


Fig. 6 Candidate positions of a staircase.

Table 1 The parameters for the example in Fig. 3.

i	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
c_i	1	1	1	4	5	3	0	1	5	7	4	3	0	0	–
d_i	3	3	3	7	6	5	3	3	7	8	7	5	3	3	2
δ_i	2	2	6	2	0	0	3	6	3	0	1	0	3	2	–
T_i	A	A	A	B	B	B	A	A	A	B	A	B	A	B	–

Table 2 Type of rectangle r_i and lowest candidate d_i.

Type of r_i	d_i
Type a	$c_i + 2$
Type b	$c_i + 3$
Type c	$c_i + 2$
Type d	$c_i + 1$ if r_i lies on the x-axis; otherwise $c_i + 2$
Type e	$c_i + 2$ if r_i lies on the x-axis; otherwise $c_i + 3$
Type f	$c_i + 1$ if r_i lies on the x-axis; otherwise $c_i + 2$

same as those for ordinary rectangular drawings in Ref. [7].

Note: In fact, that symbol 0 appears most frequently in a representation. This means that the code can be more compact by using standard data compression techniques rather than simply replacing A and B by 10 and 11, respectively (See Ref. [7] for a similar argument).

3.2 The upper bound ($5f - n_4 - 6$) of the bit length

In this subsection, we give a proof of the upper bound $5f - n_4 - 6$ of the bit length for $f \geq 2$. Consider a string representation $w = \{0, A, B\}^*$ of a rectangular drawing S_f.

Symbols A and B collectively appear exactly $f - 1$ times in w corresponding to $f - 1$ rectangles $r_2, \ldots, r_n$. They contribute exactly $2(f - 1)$ to the bit length of the corresponding code.

The number of 0's in w is equal to the sum $\sum_{i=2}^{f} \delta_i$. Now consider adding rectangular r_i to S_{i-1} at the candidate position c_i. The lowest candidate d_i of the resultant staircase S_i is at most $c_i + 3$. Precisely, d_i depends on the type of r_i (**Table 2**).

Note that if the top-left corner of r_i is a vertex of degree four, the type of r_i is c or f and $d_i = c_i + 2$. Then,

$$\begin{aligned}
\sum_{i=2}^{f} \delta_i &= \sum_{i=2}^{f} (d_{i-1} - c_i) \\
&= d_1 - c_f + \sum_{i=2}^{f-1} (d_i - c_i) \\
&\leq 2 + \sum_{i=2}^{f-1} (d_i - c_i) \qquad [d_1 = 2; c_f = 0 \text{ or } 1] \\
&\leq 2 + 3(f - 2) - n_4 = 3f - n_4 - 4.
\end{aligned}$$

Therefore, the total bit length of w is at most $2(f - 1) + (3f - n_4 - 4) = 5f - n_4 - 6$.

Now we summarize the above argument as follows.

Theorem 1 There exists an encoding of general rectangular drawings with $f (\geq 2)$ rectangles and n_4 vertices of degree four in at most $5f - n_4 - 6$ bits.

4. Concluding Remarks

In this paper, a $(5f - n_4 - 6)$-bit representation of a general rectangular drawing with $f \geq 2$ is introduced. The length of a code is at most $5f - n_4 - 6$, which is the most compact encoding ever known.

References

[1] Yao, B., Chen, H., Cheng, C.K. and Graham, R.: Floorplan Representations: Complexity and Connections, *ACM Trans. Design Automation of Electronic Systems*, Vol.8, No.1, pp.55–80 (2003).

[2] Zhuang, C., Sakanushi, K., Jin, L. and Kajitani, Y.: An Extended Representation of Q-sequence for Optimizing Channel-Adjacency and Routing-Cost, *Proc. 2003 Asia and South Pacific Design Automation Conference* (*ASP-DAC 2003*), pp.21–24 (2003).

[3] Zhao, H.A., Liu, C., Kajitani, Y. and Sakahushi, K.: EQ-Sequences for Coding Floorplans, *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, Vol.E87-A, No.12, pp.3244–3250 (2004).

[4] Yamanaka, K. and Nakano, S.: Coding Floorplans with Fewer Bits, *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, Vol.E89-A, No.5, pp.1181–1185 (2006).

[5] Fujimaki, R. and Takahashi, T.: A Surjective Mapping from Permutations to Room-to-Room Floorplans, *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, Vol.E90-A, No.4, pp.823–828 (2007).

[6] Inoue, Y., Fujimaki, R. and Takahashi, T.: Counting Rectangular Drawings or Floorplans in Polynomial Time, *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, Vol.E92-A, No.4, pp.1115–1120 (2009).

[7] Takahashi, T., Fujimaki, R. and Inoue, Y.: A $(4n - 4)$-Bit Representation of a Rectangular Drawing or Floorplan, Lecture Notes in Computer Science, Ngo, H.Q. (Ed.): *COCOON 2009*, LNCS 5609, pp.47–55 (2009).

[8] Saito, M. and Nakano, S.: Two Compact Codes for Rectangular Drawings with Degree Four Vertices, *Proc. 11th Forum on Information Technology* (*FIT 2012*), Vol.1, pp.1–8 (2012).

[9] Sakanushi, K. and Kajitani, Y.: The quarter-state sequence (Q-sequence) to represent the floorplan and applications to layout optimization, *Proc. IEEE Asia Pacific Conference on Circuits and Systems* (*APCCAS 2000*), pp.829–832 (2000).

[10] Hong, X., Huang, G., Cai, Y., Gu, J., Dong, S., Cheng, C.K. and Gu, J.: Conner Block List: An Effective and Efficient Topological Representation of Non-Slicing Floorplan, *Proc. 2000 IEEE/ACM International Conference on Computer-Aided Design* (*ICCAD 2000*), pp.8–12 (2000).

[11] Ackerman, E., Barequet, G. and Pinter, R.Y.: On the Number of Rectangular Partitions, *Proc. ACM-SIAM Symposium on Discrete Algorithms* (*SODA 2004*), pp.736–745 (2004).

[12] Murata, H., Fujiyoshi, K., Watanabe, T. and Kajitani, Y.: A mapping from sequence-pair to rectangular dissection, *Proc. 1997 IEEE Asia and South Pacific Design Automation Conference* (*ASP-DAC '97*), pp.625–633 (1997).

Editor's Recommendation

The authors improves the length of the code representing rectangular drawings, possibly containing vertices of degree four.

(Chairman of FIT2013 Ken-ichi Arakawa)

Toshihiko Takahashi received his B.E., M.E., and D.E. degrees from Tokyo Institute of Technology, Japan, in 1985, 1988, and 1991, respectively. He is currently an Associate Professor at the Institute of Natural Science and Technology, Academic Assembly, Niigata University, Japan. His current research interests include discrete mathematics and graph algorithms.