An Algorithmic Approach for Constructing Floor Plans with Circulations

Krishnendra Shekhawat, Vinod Kumar

Department of Mathematics, Birla Institute of Technology and Science, Pilani Campus, Pilani, India

Abstract. In the architectural design process, one of the preliminary steps is to build floor plans along with circulations. Here, we are the first one to present an algorithm that produces a floor plan corresponding to any adjacency requirements provided by the user/architect/designer.
After building a floor plan corresponding to given adjacency requirements, we present an algorithm for inserting spanning circulations within the obtained floor plans. Here, the spanning circulation represent the circulation space that passes through all the rooms and is adjacent to the exterior.
The presented work can be seen as a part of a larger work where the aim is to build a generic system to generate floor plans automatically, i.e., we aim to provide design tools to architects that can generate feasible solutions and can be further improved by them.

Keywords 1 algorithm, adjacency, circulation, graph theory, rectangular floor plan.

1 Introduction and Literature Review

One of the essential components of architectural design process is the designing of floor plans corresponding to given adjacency requirements, where the adjacency constraints are given in the form of an adjacency graph (in other words, an *adjacency graph* provides specific neighborhood among the given rooms). Mathematically, the problem is to map a graph into rooms such that they form a floor plan. In particular, we are interested in the mapping of an adjacency graph into rectangular rooms so that they cover a rectangular plan.

In 1964, Levin [1] was the fist one to give an intuitive idea for mapping adjacency graphs to architectural floor plans. In 1973, Steadman [2] enumerated all topologically distinct rectangular floor plans up to six rooms. Here, the problem of construction (of a floor plan) got reduced into a problem of selection among all possible solutions.

In 1977, Lynes [3] showed that if outer-planar graphs would be considered as adjacency graphs, then it is always possible to generate floor plans in which all rooms have a window. In 1980, Baybars and Eastman [4] choose a particular class of graphs, i.e., maximal planar graphs (MPG)[1], as adjacency graphs and present a procedure for constructing floor plans for the given graphs. In 1982, Baybars [5] was the first one to generate floor plans with circulation spaces (here, author considered MPG as adjacency graph).

In 1987, Rinsma [6] used graph theoretic tools for discussing the existence of rectangular floor plans corresponding to maximal outer-planar graphs. In 1988, Rinsma [7] gave conditions for the existence of rectangular and orthogonal floor plans for a given tree. In 2000, Recuero et al. [8] demonstrated a heuristic method for checking the existence of a rectangular floor plan for given adjacency requirements.

In 2010, Marson and Musse [9] proposed a method for generating dimensioned floor plans, without considering adjacency requirements, which was based on squarified treemaps algorithm[2]. In 2011, Jokar and Sangchooli [10] introduced the concept of area of a face of a graph for constructing orthogonal floor plans corresponding to maximal planar graphs. In the same year, Zhang and Sadasivam [11] studied adjacency-preserving transformations from maximal planar graphs (for which rectangular floor plan do not exist) to properly planar triangulated graphs (for which rectangular floor plan always exist). In 2012, Regateiro et al. [12] proposed an approach known as orthogonal compartment placement for architectural layout design problems, which is based on topological algebras and constraint satisfaction techniques. In 2014, Shekhawat [13] proposed the enumeration of floor plans for a particular class of planar triangulated graphs. These floor plans are known as best connected rectangular floor plans. In 2015, Shekhawat [14] presented an algorithmic approach for constructing plus-shaped floor plans and introduced the concept of co-variants for comparing and characterizing two architectural plans. In 2018, Shekhawat and Duarte [15] introduced the concept of maximal rectangular floor plans and presented their construction. In the same year, Wang et al. [16] developed a prototype to reproduce well-known legacy floor plans and to adjust or modify them so that they suit modern lifestyles while retaining the room adjacencies.

It can be observed from above discussion that the existence of a rectangular floor plan for the given adjacency requirement is not always possible $[6-8]$, at the same time, enumerating all rectangular floor plans for the given number of rooms is very exhaustive $[2, 13, 15]$. To address these issues, in this paper, we first propose an algorithm for enumerating maximal rectangular floor plans, i.e., a set of rectangular floor plans (maximal) that topologically contains all rectangular floor plans. The obtained maximal rectangular floor plan can be further reduced into a floor plan corresponding to given adjacency constraints. Then, we present an algorithm for inserting circulations within the obtained floor plans.

---

[1] A planar graph G is *maximal* if no edges can be added to G without losing planarity.

[2] Treemaps subdivide an area into small pieces to represent the importance of each part in the hierarchy whereas squarified treemaps are used to generate rooms with aspect ratios close to one.

In this paper, we aim to tackle a problem that has been inspired by architecture of allocating rectangular spaces within a rectangular framework with certain combination of rectangular spaces designated as circulation spaces.

1.1 Terminologies

In this section, we first discuss few terminologies and notations which are frequently used in this paper. For further detail related to most of the definitions, refer to [13, 15].

In a *rectangular floor plan* (RFP) each room is rectangle along with its boundary. In an *orthogonal floor plan* (OFP) the boundary is rectangular while the rooms are rectilinear, i.e., made up of more than one rectangles.

For each rectangular floor plan, a graph can be constructed called *dual graph* DG_n, where each room is replaced by a vertex and two vertices are adjacent if corresponding rooms are adjacent. For example, the dual graph of a rectangular floor plan in Figure 1A is shown in Figure 1B.

A graph which is the dual graph of a rectangular floor plan is called *rectangular floor plan graph*, abbreviated as $\mathrm{RFP_G}$. For example, the graph shown in Figure 1C is a $\mathrm{RFP_G}$ because there exist a rectangular floor plan corresponding to it as shown in Figure 1D while the graphs in Figures 1F and 1G are not $\mathrm{RFP_G}$ (it is not possible to construct a rectangular floor plan corresponding to them).

Two rectangular floor plans with same number of rooms are said to be *distinct* or *non-isomorphic* if their dual graphs are non-isomorphic. For example, the rectangular floor plans in Figures 1A and 1D are distinct (rooms R_2 and R_5 are adjacent in Figure 1D but they are not adjacent in Figure 1A) while the floor plans in Figures 1D and 1E are isomorphic because they preserve all adjacencies.

Two rectangular floor plans are said to be *topologically equivalent* if and only if one can be derived from the other using translation, rotation, reflection and scaling. Conversely, two rectangular floor plans are *topologically distinct* if they are not topologically equivalent. For example, it is interesting to note that the rectangular floor plans in Figures 1D and 1E are topologically distinct but isomorphic (one can not be transformed into the other by using any of the transformations but they have isomorphic dual graphs).

A $\mathrm{RFP_G}$ is called a *maximal* $\mathrm{RFP_G}$, denoted as $\mathrm{MRFP_G}$, if adding a new edge to it produces a graph that is not a $\mathrm{RFP_G}$. A RFP corresponding to a $\mathrm{MRFP_G}$ is called *maximal rectangular floor plan* (MRFP). The $\mathrm{RFP_G}$ in Figure 1C is a $\mathrm{MRFP_G}$ because it is not possible make any two non-adjacent rooms in Figure 1C adjacent while preserving rectangularity of rooms and floor plans along with the inbuilt adjacencies.

For convenience, all distinct $\mathrm{MRFP}(n)$ for a fixed number of rooms, say n, can be grouped under a same umbrella and called them as *generic rectangular floor plans* with n rooms, abbreviated as $\mathrm{GRFP}(n)$.

A $\mathrm{RFP}(n)$ is called *best connected* if its dual graph has $3n-7$ edges and it is denoted as BCRFP. The dual graph of a BCRFP is called best connected $\mathrm{RFP_G}(n)$, represented as $\mathrm{BCRFP_G}(n)$. For example, Figure 1C is $\mathrm{BCRFP_G}(5)$ (because it has $3n-7 = 3\times5-7 = 8$ edges and there exists a RFP corresponding

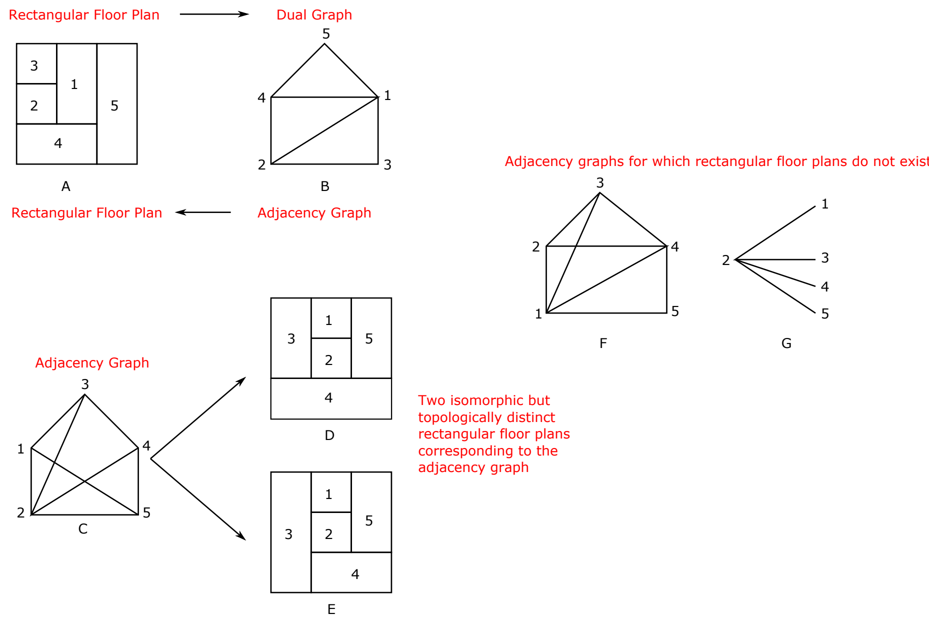

Fig. 1. Demonstrating the concepts that are frequently used in the paper

to it in Figure 1D), and Figures 1D-1E are BCRFP(5) (because their dual graph in Figure 1C has $3n - 7$ edges).

1.2 Why Generic Rectangular Floor Plans

The well-known problem related to architectural design is to construct a rectangular floor plan (RFP) corresponding to a given adjacency graph (an adjacency graph is a representation of the connections between the rooms; RFP stands for a floor plan with all rooms rectangular). For an example, refer to Figures 2A and 2B illustrating a given adjacency graph and corresponding RFP respectively.

It should be noted that there may or may not exists a RFP corresponding to a given adjacency graph. For example, there exists a RFP for the graphs in Figure 2A but it is not possible to construct a RFP for the graph in Figures 2C and 2D.

To understand the concept of generic floor plans, let us restrict ourselves to the RFP having four rooms. It is easy to verify that there exist only six connected graphs with four vertices (refer to Figures 2C to 2H). Now, we present a floor plan called generic floor plan which satisfies adjacency requirements of all the six graphs in Figures 2C to 2H and from which a floor plan corresponding to them can be derived.

In the Table in Figure 2, for the first 5 graphs, a floor plan in the last column is the generic rectangular floor plan, where a rectangular floor plan does not exist for the graphs in fifth and sixth row. For the graph in sixth row, even a generic rectangular floor plan does not exist, hence we derive an orthogonal floor plan corresponding to it as shown in Figures 2I to 2L.

Therefore, from Figure 2, it is clear that, it is possible to construct a generic floor plan for all possible adjacency graphs.

As the number of rooms (n) increases, the number of adjacency graphs for a fixed n is very large (for an illustration, see Table 1). Also, there does not exist any algorithm for constructing a floor plan for any given arbitrary graph (there only exist some work for constructing floor plans for planar triangulated graphs [4, 5, 10, 13]). In this case, first we do not know if there exists a RFP for a given graph, and if it exists, there does not exist any algorithm for its construction. Therefore, in this paper, we propose to construct generic rectangular floor plans (a set of all distinct maximal rectangular floor plans) because any graph for which a RFP exists must be a sub-graph of a maximal RFP, i.e., the adjacency requirements for any graph for which it is possible to construct a RFP can be satisfied using generic RFP. In addition to it, corresponding to the graphs for which RFP do not exist, it is possible to construct a maximal RFP or an orthogonal floor plan using generic RFP (for the graph in Figure 2C and 2D, a RFP do not exist, therefore, a maximal RFP and an orthogonal floor plan are constructed using generic RFP, as shown in the Table in Figure 2 and Figures 2I to 2L). Hence, using the generic RFP, we can construct a floor plan for all the connected graphs.

Table 1. The number of connected graphs

Number of vertices	Number of connected graphs
1	1
2	1
3	2
4	6
5	21
6	112
7	853
8	11117
9	261080
10	11716571
11	1006700565
12	164059830476
13	50335907869219
14	29003487462848061
15	31397381142761241960
16	63969560113225176176277

1.3 Work Done

In 2018, Shekhawat [17] gave an algorithmic approach for the enumeration of generic rectangular floor plans (GRFP). In this paper, we first present an algorithm for constructing all distinct maximal rectangular floor plans (MRFP)

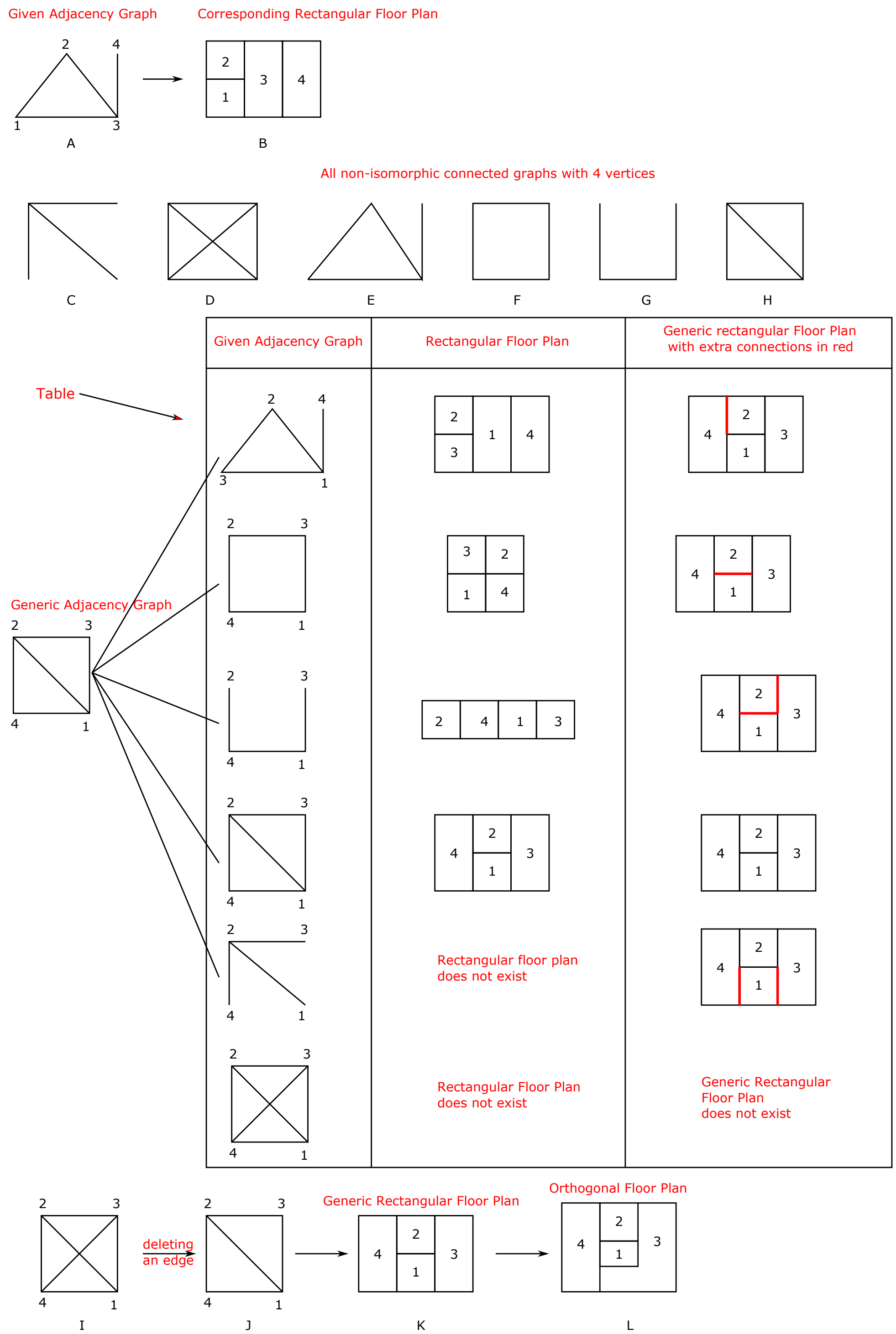

Fig. 2. Constructing a generic rectangular floor plan for all possible adjacency requirements when the number of rooms is four

using the concept of rectangular dissection, which is far more simpler than the approach used in [17]. After constructing the required GRFP, we present an algorithm to derive a floor plan from a MRFP corresponding to any adjacency requirements. Then we present an algorithm for inserting circulation spaces within the obtained RFP.

2 Enumerating Generic Rectangular Floor Plans

It can be found in the literature that a rectangular floor plan (RFP) can be constructed in the following two ways:

- Addition, where rooms are added one by one for producing a RFP [18],
- Dissection, where a larger rectangle is dissected into the smaller rectangular pieces called rooms [19, 20].

Recently, Shekhawat [17] presented the enumeration of generic rectangular floor plans (GRFP) by adding rooms. In this section, we present an algorithm for the construction of GRFP using rectangular dissection, which is conceptually simpler and involves less computation than the algorithm proposed by [17].

Shekhawat [17] also gave the following theorem for characterizing the maximal rectangular floor plans (MRFP).

Theorem 1. A $\mathrm{MRFP}_{\mathrm{G}}(n)$ is either a $\mathrm{BCRFP}_{\mathrm{G}}(n)$ (best connected rectangular floor plan graph) or W_n (*wheel graph* formed by connecting a single vertex to all vertices of a cycle).

From Theorem 1, it is clear that $\mathrm{GRFP}(n)$ can be derived by enumerating $\mathrm{RFP}(n)$ corresponding to W_n and $\mathrm{BCRFP}_{\mathrm{G}}(n)$. It is easy to see that for a fixed n, the wheel graph W_n is unique but there may exists many non-isomorphic $\mathrm{BCRFP}_{\mathrm{G}}(n)$ for a fixed n. Hence, the problem here is to enumerate all non-isomorphic $\mathrm{BCRFP}_{\mathrm{G}}(n)$. In [15], authors have presented a techniques for enumerating all distinct $\mathrm{BCRFP}_{\mathrm{G}}(n)$ but it is very exhaustive. Therefore, in this section, instead of enumerating $\mathrm{BCRFP}_{\mathrm{G}}(n)$, we propose an algorithm for enumerating $\mathrm{BCRFP}(n)$, which is comparatively simpler than the algorithms proposed in [15, 17]. Before moving to the algorithm, we first present a simple procedure for constructing a RFP corresponding to W_n as follows (it is important to note here that a RFP for any graph is not unique, i.e., there may exist many topologically distinct RFP for a given graph):

Begin the construction of a RFP for W_n by drawing the first room, say R_A. Then draw the remaining $n-1$ rooms in such a way that:

- all $n-1$ rooms must be adjacent to R_A,
- among $n-1$ rooms, there should be at least one room adjacent to each side of R_A.

This gives us the required MRFP corresponding to W_n. As an example, W_6 and its corresponding MRFP are illustrated in Figure 3.

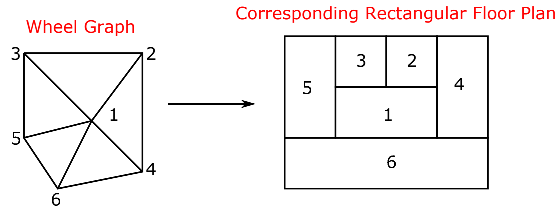

Fig. 3. A wheel graph with six vertices and its corresponding rectangular floor plan

From above discussion it is clear that to have GRFP(n) for a fixed n, we need to derive all distinct MRFP(n), i.e., we mainly need to construct all distinct BCRFP(n). Now, we present a recursive algorithm for computing all distinct BCRFP(n) from all distinct BCRFP($n-1$), where $n > 4$. To have this recursive algorithm, we first require all distinct BCRFP(5) ,which have already been computed in [15] and shown in Figure 4A1.

Algorithm 1 *MRFP Recursive Algorithm*

The steps of the algorithm are as follows:

1. Deleting the boundary rooms
   Consider all (say r) BCRFP($n-1$) and delete all four boundary rooms of each BCRFP($n-1$). It gives us r distinct RFP($n-5$).
   Example. After deleting the boundary rooms of BCRFP(5), RFP(1) is shown in Figure 4A2.
2. Horizontal and vertical dissection of rooms
   Consider each room of the obtained RFP($n-5$), one at a time, and dissect it horizontally to have one or more RFP($n-4$). Similarly, each room of RFP($n-5$) is vertically dissected to derive remaining RFP($n-4$). This step gives at least $2(n-5)$ RFP($n-4$).
   Example. The room of RFP(1) (see Figure 4A2) can be dissected horizontally and vertically only in one way to have two RFP(2) as shown in Figures 4A3 and 4A4 respectively. Similarly, each room of RFP(2) in Figure 4B1 (obtained by deleting boundary rooms of BCRFP(6) in Figure 4A6) is dissected horizontally and vertically respectively to have four RFP(3) as shown in Figures 4B2 to 4B5 respectively.
3. Identifying topologically distinct RFP($n-4$)
   To avoid repetition, among all the obtained RFP($n-4$), choose all topologically distinct RFP($n-4$).
   Example. For $n-4=2$, both of the RFP($n-4$) in Figures 4A3 and 4A4 are topologically equivalent. Hence, it pick any one of them. Similarly, for $n-4=3$ (see Figures 4B2 to 4B5), there are two topologically distinct RFP($n-4$), i.e., we pick RFP($n-4$) in Figures 4B2 and 4B3. Also, for $n-4=4,5$, there are six and twenty topologically distinct RFP($n-4$), as illustrated in Figures 4C3-4C8 and 5B1-5B20 respectively.

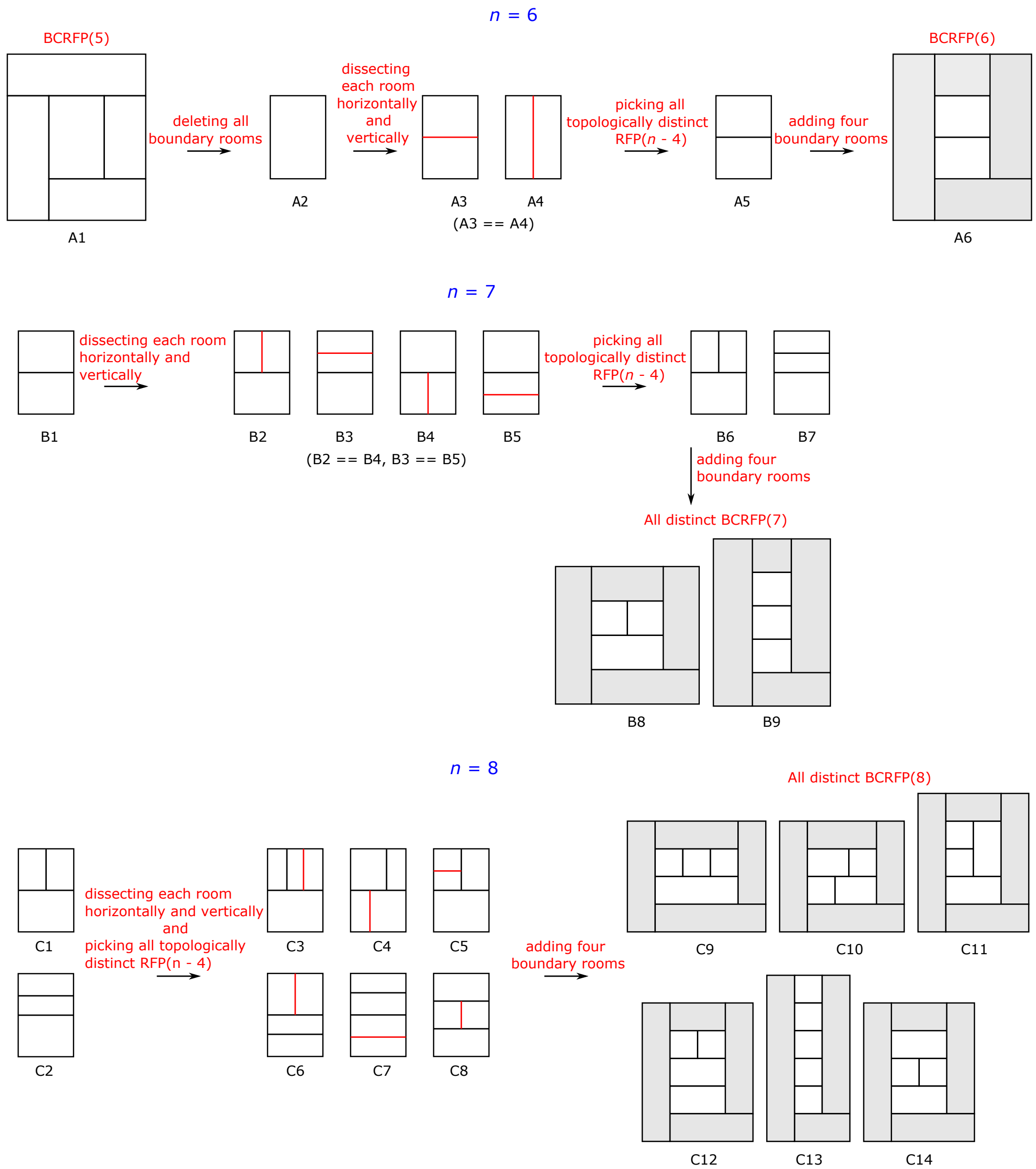

Fig. 4. Computing all distinct best connected rectangular floor plans for n rooms where $n = 6, 7, 8$ (here $==$ symbol represents two topologically equivalent rectangular floor plans)

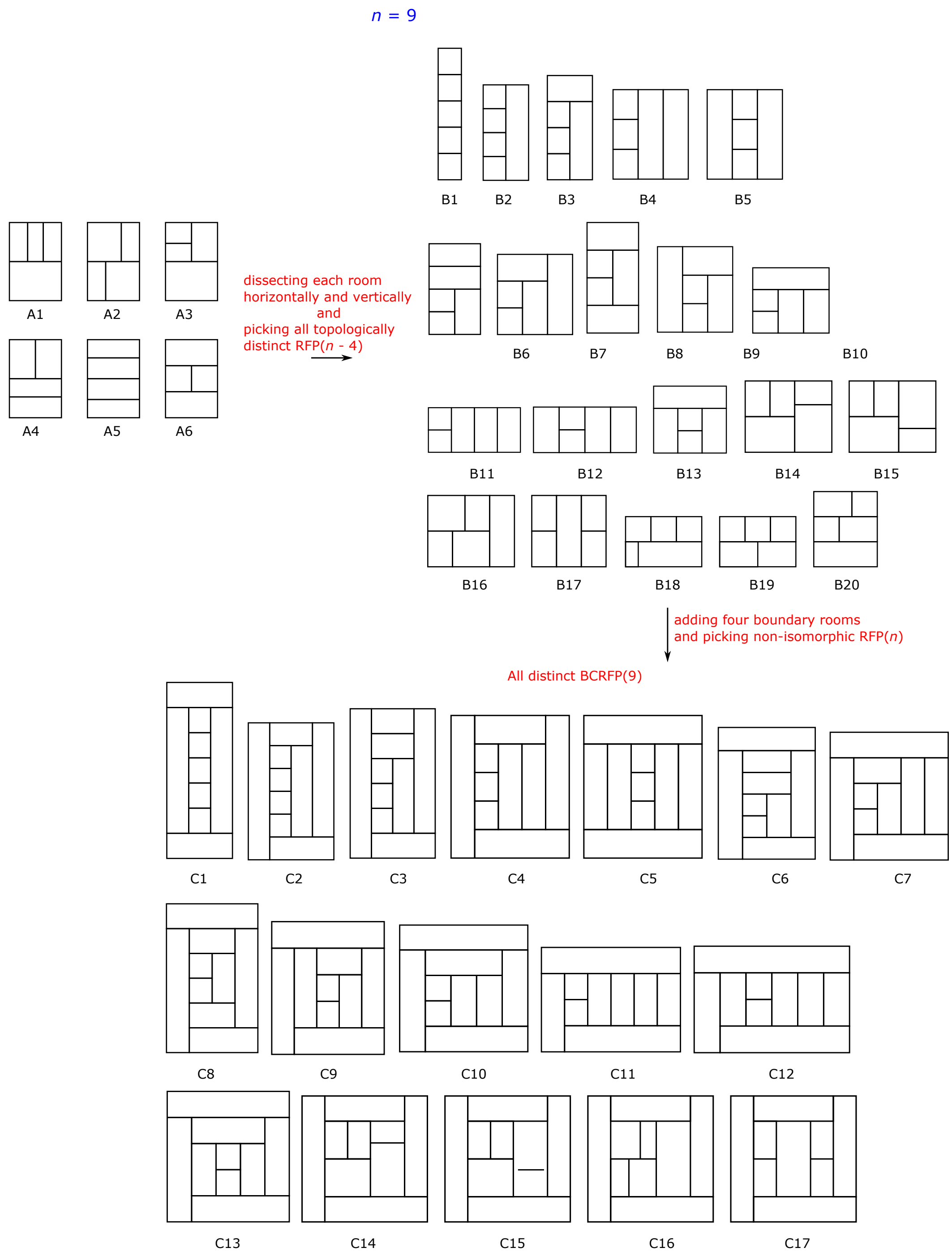

Fig. 5. Computing all distinct best connected rectangular floor plans for nine rooms

4. Obtaining BCRFP(n) from RFP($n-4$)
   Add a rectangular room to each side of the obtained RFP($n-4$) in Step 3 to have, say s, BCRFP(n) (four boundary rooms are added to the four sides of RFP($n-4$) respectively such that after adding each room the overall composition of the rooms should be rectangular).
   Example. BCRFP(n) corresponding to RFP($n-4$) for $n = 6, 7, 8$ in Figure 4 are obtained by adding boundary rooms to all corresponding RFP($n-4$) (see Figure 4A6 for BCRFP(6); 4B8, 4B9 for all BCRFP(7); 4C9-4C14 for all BCRFP(8)).
5. Deriving distinct BCRFP(n)
   To eliminate isomorphic BCRFP(n), among s BCRFP(n) obtained in Step 4, derive the distinct BCRFP(n) (for picking distinct BCRFP(n), we need to first obtain all distinct $\mathrm{BCRFP_G}(n)$, which can be done using the algorithm given by Lee et al. [21]).
   Example. For $n = 9$, there are 20 topologically distinct BCRFP(9) among which only 17 BCRFP(9) are distinct as shown in Figure 5. For $n = 6, 7, 8$, all BCRFP(n) obtained in Step 4 are distinct.

Remark 1. In Step 4 of above algorithm, we used a method for constructing BCRFP(n) which is based on the result that a RFP(n) is best connected if and only if it has only four rooms on the boundary. This result will be proved in next section (see Theorem 2).

2.1 Deriving a Floor plan from a MRFP for Given Adjacency Requirements

Once we have all MRFP(n), we can derive a floor plan for any given graph G_n using the obtained MRFP. The steps involved in obtaining a floor plan corresponding to a given graph G_n are as follows (for the demonstration of the steps involved in this algorithm, refer to the flow chart in Figure 6):

Algorithm 2 1. If G_n is a sub-graph of any of the $\mathrm{MRFP_G}(n)$
   If G_n is not a sub-graph of any of the $\mathrm{MRFP_G}(n)$, it means that it is not possible to construct a rectangular floor plan (RFP) for G_n (it directly follows from Theorem 1). In this case, we need to construct an orthogonal floor plan (OFP) for G_n. If G_n is a sub-graph of a $\mathrm{MRFP_G}(n)$, then there may or may not exist a RFP for G_n. In this case, we first try to construct a RFP, if it exists, otherwise we construct an OFP for G_n.

   (a) G_n is a sub-graph of a $\mathrm{MRFP_G}(n)$
   All distinct $\mathrm{MRFP_G}(n)$ can be obtained using the Algorithm 1. Hence, they can be stored and we can easily identify the $\mathrm{MRFP_G}(n)$ which is a super-graph of G_n.
   Example. The graph G_6 in Figure 7A is the given graph and it is a sub-graph of $\mathrm{MRFP_G}(6)$ in Figure 7B.

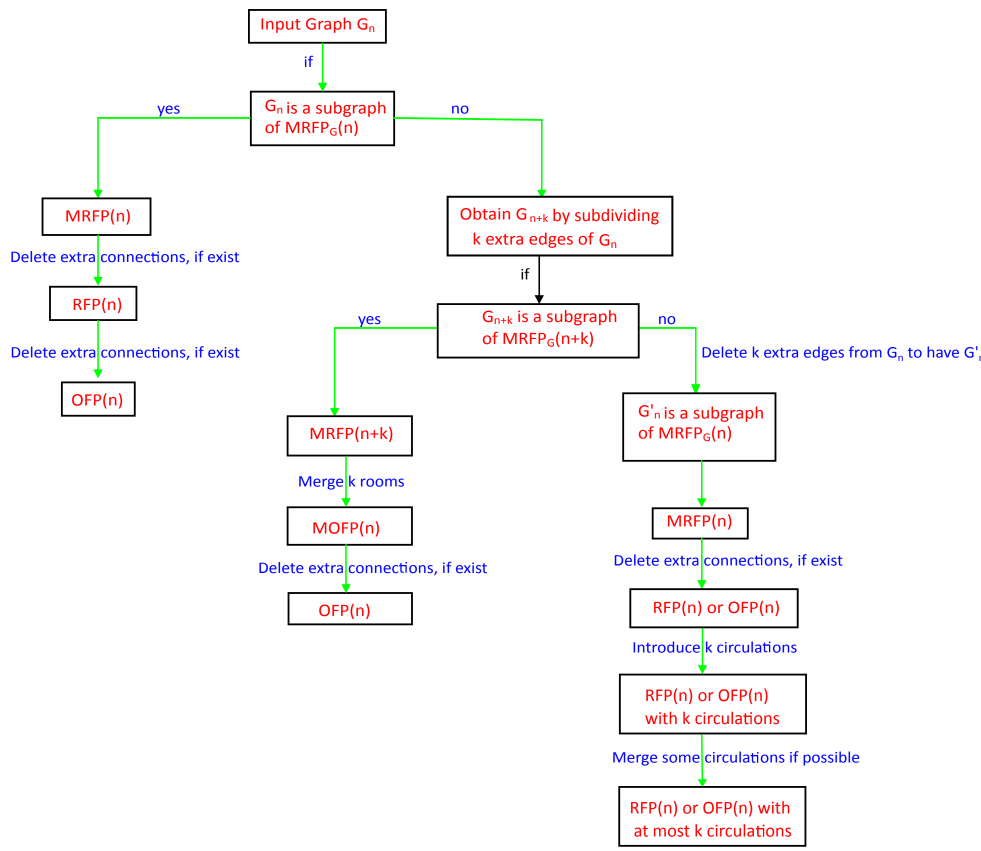

Fig. 6. Flow chart illustrating the different cases for constructing a floor plan corresponding to a given adjacency graph

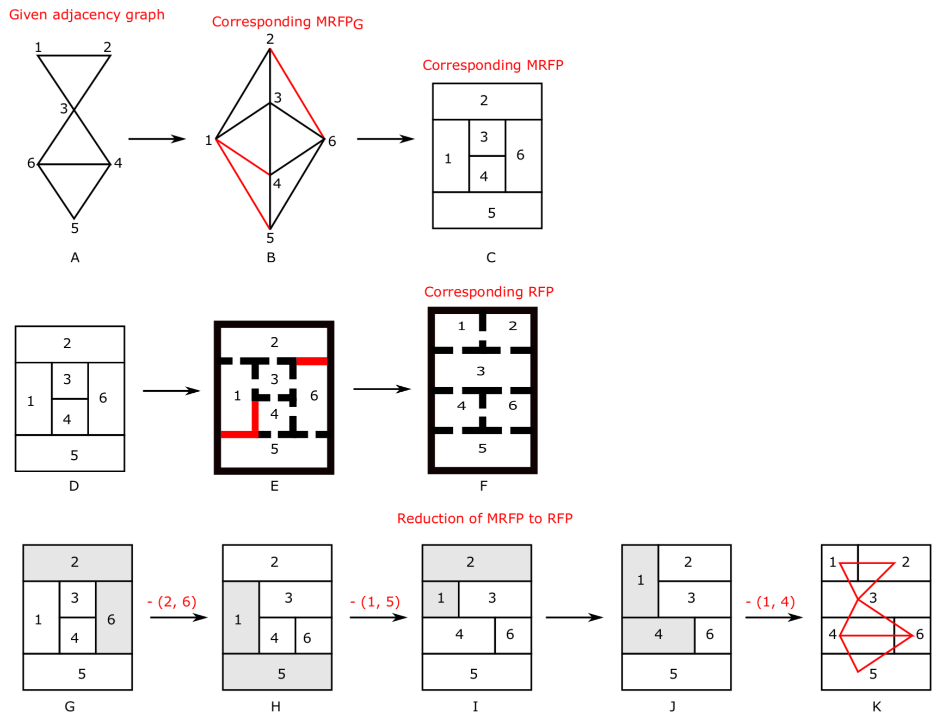

Fig. 7. Reducing the maximal rectangular floor plan corresponding to a given adjacency graph into a rectangular floor plan for given graph by eliminating the extra connections (in Figure 7B and 7E, red edges and red walls represent the extra connections respectively)

(b) Deriving MRFP(n) for G_n

If G_n is a sub-graph of a $\mathrm{MRFP_G}(n)$, we derive the required MRFP(n), which is a RFP for G_n with some extra connections.

Example. MRFP(6) in Figure 7C is the required RFP with extra connections between rooms, illustrated by red color in Figure 7B.

2. If G_n is not a sub-graph of any of the $\mathrm{MRFP_G}(n)$

(a) Deriving a sub-graph G'_n of a $\mathrm{MRFP_G}(n)$ from G_n

Delete minimum number of edges of G_n such that their deletion results in a graph G'_n which is a sub-graph of at least one of the $\mathrm{MRFP_G}(n)$ (these minimum number of edges can be computed using the algorithm presented in Section 2.2).

Example. The graph G_8 in Figure 9A is not a sub-graph of any of the $\mathrm{MRFP_G}(8)$ but deletion of two edges joining v_1 and v_7, v_3 and v_5 results in a graph that would be a sub-graph of at-least one of the $\mathrm{MRFP_G}(8)$.

(b) Subdividing the edges of G_n whose deletion formed G'_n

Subdivide each of the edges of G_n, whose deletion formed G'_n, into two edges, by adding a vertex to each of them, to have a new graph, say G_r.

Example. Two new vertices v_9 and v_{10} have been inserted to the graph in Figure 9A to have a new graph shown in Figure 9B.

3. If G_r is a sub-graph of a $\mathrm{MRFP_G}(r)$

Here, G_r may or may not a sub-graph of a $\mathrm{MRFP_G}(r)$. If the graph G_n is planar, then for sure G_r is a sub-graph of a $\mathrm{MRFP_G}(r)$ and in this case, we obtain a maximal orthogonal floor plan for G_n. If G_r is not a sub-graph of any of the $\mathrm{MRFP_G}(r)$, it means the graph G_n is not planar. It is very important and interesting to note here that we can also check the planarity of a graph using this algorithm. Even, in the literature, many authors used the word planar floor plans instead of floor plans because floor plans can be constructed for planar graphs only. Here, we are the first one to propose the constructing of floor plans for non-planar graphs. It would be done by introducing some circulation spaces.

(a) Obtaining a MRFP for G_r

Since G_r is a sub-graph of a $\mathrm{MRFP_G}(r)$, obtain a $\mathrm{MRFP_G}(r)$ for G_r and construct the corresponding MRFP(r).

Example. A $\mathrm{MRFP_G}(10)$ and MRFP(10) for G_{10} in Figure 9B are shown in Figures 9C and 9D respectively.

Let e_{ij} denotes an edge with endpoints as vertices v_i and v_j and vertex v_p is the added vertex.

(b) Reducing obtained MRFP(r) into a maximal orthogonal floor plan (MOFP) for G_n

Corresponding to each new vertex v_p, $n < p \leq r$, merge room R_p into either room R_i or room R_j, to have a maximal orthogonal floor plan (MOFP) for G_n.

Example. Corresponding to G_8 in Figure 9A, a MOFP is demonstrated in Figure 9E which has been obtained from MRFP(10) in Figure 9D by merging room R_9 into room R_3, and room R_{10} into R_7.

4. If G_r is not a sub-graph of any of the $\mathrm{MRFP_G}(r)$
   Example. G_7 in Figure 8B (obtained from G_5 in Figure 8A by introducing extra vertices) is not a subgraph of any of the $\mathrm{MRFP_G}(7)$.
   (a) Consider the graph G'_n (obtained by deleting k edges in Step 2a, Algorithm 2).
   Example. G'_5 for G_5 in Figure 8A is shown in Figure 8C.
   (b) Obtain a MRFP(n) for G'_n.
   For example, a MRFP for G'_5 in Figure 8C is illustrated in Figure 8D.
   (c) Eliminate the extra connections, if exist in the obtained MRFP.
   Example. There does not exist extra connections in the MRFP corresponding to G'_5 in Figure 8C.
   (d) Introduce k circulations corresponding to the deleted edges. If possible, insert circulations such that they are adjacent to exterior.
   Example. Two circulations have been introduced in Figure 8E corresponding to the edges deleted from G_5 in Figure 8A.
   (e) Merge circulations with the rooms, if possible, while preserving all adjacencies.
   Example. A circulation has been merged to room R_3 to have a required floor plan for G_5 in Figure 8A, as shown in Figure 8F. For the floor plan in Figure 8F, R_1 and R_5, R_2 and R_3, are adjacent through circulation and it is not possible to merge this circulation in any of the room while preserving other adjacencies.
5. Reducing the obtained maximal floor plan into a floor plan for G_n
   Eliminate all extra connections from the obtained maximal floor plan in Step 1 or Step 3 to have a required floor plan for G_n.
   Example. Figures 7K and 9H illustrate the required floor plan corresponding to the graphs in Figures 7A and 9A respectively. The extra connections are represented by red edges in the corresponding $\mathrm{MRFP_G}(n)$ or $\mathrm{MOFP_G}(n)$, which are eliminated by deleting the corresponding connections in respective MRFP or MOFP, as shown in Figures 7G-7K and 9I-9L respectively.

2.2 Computing Extra Edges

From Algorithm 2, we have seen that if there does not exist a RFP for a given adjacency graph G_n, we need to construct an OFP corresponding to it. In this case, we need to compute k, where k is the minimum number of edges of G_n whose deletion results in a graph G'_n which is a sub-graph of at least one of the $\mathrm{MRFP_G}(n)$. The steps for computing k are as follows:

1. Let $E(G_n)$ be the edge set of G_n. First compute the power set of $E(G_n)$ and denote it as $P(E(G_n))$.
2. Let the edge sets (elements) of $P(E(G_n))$ be denoted by $S_1, S_2, \ldots, S_{2^m}$. Arrange $S_1, S_2, \ldots, S_{2^m}$ in the decreasing order on the basis of size of each set S_i, where size of a set is the number of elements in the set.
   Let the elements of $P(E(G_n))$ in the descending order be $S'_1, S'_2, \ldots, S'_{2^m}$.
3. Let $i = 2$.

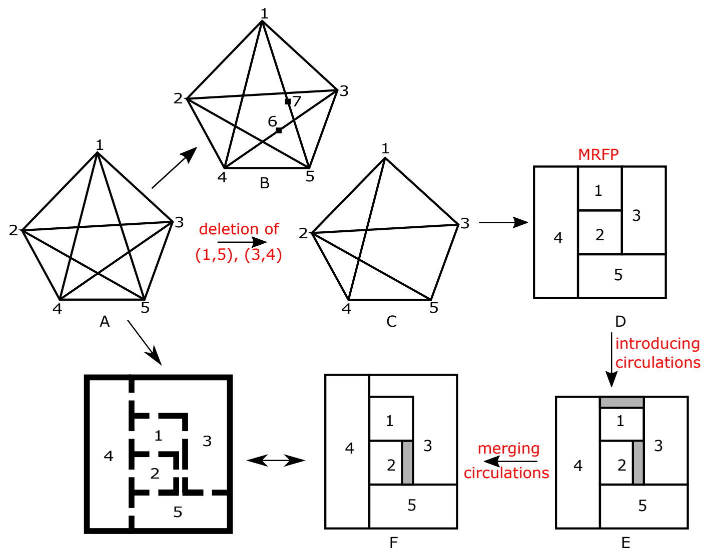

Fig. 8. Introducing circulations into a maximal rectangular floor plan to have a floor plan corresponding to a given adjacency graph (for which a floor plan does not exist)

4. Choose S'_i.
5. If $S'_i \subseteq E(MRFP_{\mathrm{G}}(n))$, then $k = m - n(S'_i)$ and edges corresponding to k are $E(G_n) - S'_i$.
6. If condition in Step 5 holds, stop the algorithm otherwise increase i by 1.
7. If $i \leq (2^m)$ go to Step 4 otherwise stop.

2.3 Best Connected Rectangular Floor Plans (BCRFP)

In this section, we will prove that a rectangular floor plan is best connected if and only if it has four rooms on the boundary. To proceed, we first introduce few terminologies as follows:

A point in a floor plan where two or more walls/sub-walls coincide is called a *joint*. Joints are further classified as k-joints where k is the number of walls/sub-walls meet at the joint. Refer to RFP(4) in Figure 10A, where a 3-joint has been marked with an arrow.

A wall or a sub-wall between any two joints is called a *wall-segment*. A wall-segment forming a part of plan boundary is called an *external wall-segment* otherwise an *internal wall-segment*. For example, in RFP(4) in Figure 10A, there are 5 internal wall-segments marked by dotted lines, while remaining 8 wall-segments are external.

Let the number of wall-segments in a RFP(n) be denoted by WS(n).

Lemma 1. For a RFP(n) without 4-joints, WS(n) $= 3n + 1$.

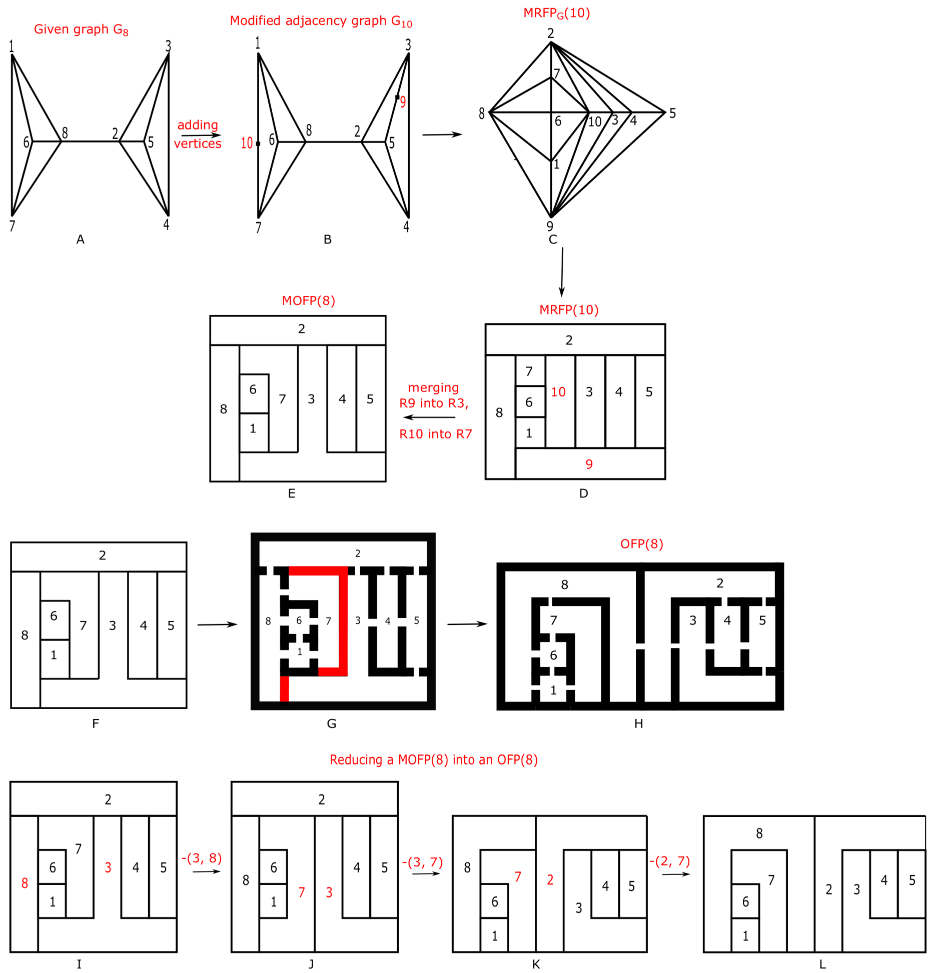

Fig. 9. Constructing a maximal orthogonal floor plan for a given adjacency graph for which maximal rectangular floor plan does exist and then reducing it into a floor plan corresponding to the given graph by eliminating the extra connections

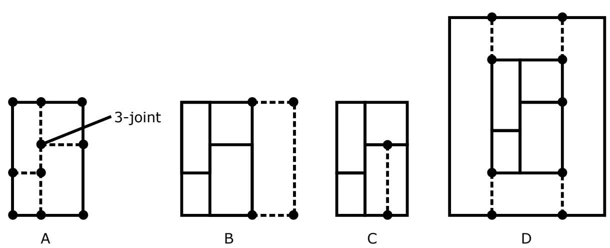

Fig. 10. Illustrating different concepts related to a best connected rectangular floor plan

Proof. We will prove the result by mathematical induction on n. For $n = 1$, the result is trivial. Let the result be true for $n = k$, i.e., $\mathrm{WS}(k) = 3k + 1$. We will show that the result is true for $n = k + 1$.

RFP$(k + 1)$ can be obtained from RFP(k) by the following two methods:

- By adding a new room to RFP(k)
  In this case, the number of wall-segments get increased by 3, as shown in Figure 10B.
- By dissecting an existing room of RFP(k)
  In this case, first we need to introduce a new wall-segment to dissect a room say R_i, such that the dissection of room R_i horizontally/vertically dissects the corresponding vertical/horizontal walls of R_i into two wall-segments. This dissection clearly increases the number of wall-segments by three. For example, for RFP(4) in Figure 10A, $\mathrm{WS}(4) = 13$ and after dissecting a room of the RFP(4), we obtain RFP(5) in Figure 10C with $\mathrm{WS}(5) = 16$.

In both the cases, $\mathrm{WS}(k+1) = \mathrm{WS}(k)+3 = 3k+1+3 = 3(k+1)+1$. Hence, the result is true for $n = k + 1$.

Theorem 2. A RFP(k) is best connected if and only if it has four rooms on the boundary.

Proof. From Lemma 1, for a RFP(k), we have $\mathrm{WS}(k) = 3k + 1$. If four rooms are added at the boundary of RFP(k), the number of internal wall-segments of RFP$(k + 4)$ would be at least $3k + 1$.

Since, all four boundary rooms are adjacent to each other, they will increase the number of internal wall-segments by four (for an illustration, see Figure 10D which is obtained from Figure 10A by adding four rooms at the boundary). Hence, the number of internal wall-segments of RFP$(k + 4)$ is $3k + 1 + 4 = 3(k + 4) - 7$, i.e., the dual graph of RFP$(k + 4)$ has $3k + 1 + 4 = 3(k + 4) - 7$ edges. This implies that the RFP$(k + 4)$ is best connected if it has four rooms on the boundary.

On the other hand, it is easy to verify that if there are more than four rooms on the boundary of RFP$(k + 4)$, then the number of external wall-segments are more than 8, i.e., the number of internal wall-segments are less than $(3k+13)-8$. Hence, its dual graph has less than $3(k + 4) - 7$ edges. Hence, RFP$(k+4)$ is not best connected.

3 Enumerating Spanning Circulations within a Rectangular Floor Plan

A *circulation* refers to the way people move through and interact within a building. A *spanning circulation space* stands for a single interior courtyard adjacent to each of the rooms of a floor plan.

In Section 2, we have presented an algorithm for constructing all distinct MRFP. In this section, we first present an algorithm for inserting spanning circulation inside a MRFP. Then we will show that this algorithm can be used for inserting circulations inside any rectangular floor plan. Here, we assume that floor plans have only one entrance.

Let f be the number number of interior faces in a $\mathrm{MRFP_G}$, represented as $F_1, F_2, \ldots, F_f$. Also, consider $E = \{e_1, e_2, \ldots, e_m\}$ and $V = \{v_1, v_2, \ldots, v_n\}$ be the edge set and vertex set of $\mathrm{MRFP_G}$ respectively. The steps of the algorithm for generating circulations are as follows (for understanding the steps of the algorithm, refer to Figures 11A and 11B, where a MRFP and corresponding $\mathrm{MRFP_G}$ are illustrated):

Algorithm 3 1. Construct the dual graph of the given MRFP, which is the required $\mathrm{MRFP_G}$ corresponding to the MRFP.
Example. Figure 11B is the $\mathrm{MRFP_G}$ corresponding to MRFP in Figure 11A.

2. Choose a face, say F_1, of $\mathrm{MRFP_G}$ having an exterior edge, say e_1 (edges adjacent to the exterior are the *exterior edges*).
Example. Corresponding to Figure 11B, we choose face F_1 that comprises of vertices v_1, v_2, v_7.
3. Consider an empty set S. Add all the three vertices that forms F_1 to S.
Example. Corresponding to Figure 11C, $S = \{v_1, v_2, v_7\}$.
4. Insert a new vertex, say V_{n+1}, on the exterior edge e_1. By inserting a new edge say E_k, make V_{n+1} adjacent to a vertex of F_1 that is not an end-point of e_1.
Example. V_8 and E_{15} are shown in Figure 11C (in Figure 11, inserted vertices are marked by red color while inserted edges are illustrated by dotted lines).
5. Insert a rectangular circulation space, say C_{n+1}, corresponding to V_{n+1} in the corresponding MRFP (the conditions for inserting a circulation space are given in Step 9).
Example. In Figure 11D, inserted circulation space C_8 is adjacent to rooms R_1, R_2 and R_7 (the inserted circulation spaces are demonstrated by grey color).
6. Choose a face, say F_i, $2 \leq i \leq f$, to which an edge has not been inserted such that F_i is adjacent to at least one of the faces to which an edge has already been inserted. Let F_i is adjacent to F_j, $1 \leq j \leq f$.
Example. In Figure 11E, we choose a face that is made up of vertices v_2, v_7, v_3 and is adjacent to F_1.
7. Pick an edge e_j that is common to both F_i and F_j. Subdivide edge e_j by inserting a new vertex, say V_k, $k \geq n+2$ and make V_k adjacent to a vertex of F_i that is not an end-point of e_j.
Example. In Figure 11E, V_9 and E_{16} have been inserted.
8. Add all three vertices that forms F_i to S.
Example. Corresponding to Figure 11E, $S = \{v_1, v_2, v_7, v_3\}$.
9. Insert a circulation space, say C_k, corresponding to V_k in the corresponding MRFP such that it preserves the adjacency relations of V_k, i.e., it only satisfies the following adjacency requirements:
   (a) C_k is adjacent to rooms corresponding to the vertices that are adjacent to V_k.
   Example. In Figure 11F, C_9 is adjacent to R_2, R_3, and R_7.

(b) C_k must be adjacent to one of the existing circulation space.
Example. In Figure 11F, C_9 is adjacent to C_8.

(c) Let the endpoints of e_j be v_j^1 and v_j^2. C_k is inserted in such a way that it makes the rooms corresponding to v_j^1 and v_j^2 non-adjacent.
Example. In Figure 11F, after inserting C_9, rooms R_2 and R_7 are non-adjacent.

(d) To satisfy adjacency requirements, C_k can be inserted horizontally or vertically.
Example. In Figure 11F, C_9 has been inserted vertically.

10. If $S \subset V$, go to Step 6 otherwise stop.
Example. Corresponding to Figure 11B, a MRFP with all inserted circulation spaces is shown in Figure 11H.
11. Adjoin all inserted circulation spaces to form a spanning circulation.
Example. MRFP with spanning circulation is shown in Figure 11I. Similarly, MRFP with different position of spanning circulations are illustrated in Figures 11J and 11K.

Remark 2. Using Algorithm 3, we can also insert circulations inside any rectangular floor plan. For example, consider an adjacency graph G_9 in Figure 12A. Using Algorithm 2, we first obtain a MRFP for G_9, as shown in Figure 12B. The obtained MRFP is further reduced into a RFP for G_9 (see Figures 12D to 12J). At the end, using Algorithm 3, we insert circulation spaces inside the RFP in Figure 12J as shown in Figure 12K.

4 Conclusion and Future Work

In this paper, we aim to produce a floor plan design for the adjacency requirements asked by the user. But for the given adjacency requirements, there may or may not exist a floor plan. And if it exist, there may or may not exist a rectangular floor plan. Also, if it exist, it is not easy to construct it, in particular when the number of rooms are large. Furthermore, constructing an orthogonal floor plan is even more complex than the construction of a rectangular floor plan.

To address all of these issues, we introduced the concept of maximal rectangular floor plans because of the following reasons:

– Any adjacency graph for which a rectangular floor plan exists is always a sub-graph of a maximal rectangular floor plan, which can be computed using Algorithm 1 and hence, stored in advance. If it is computationally demanding to have a rectangular floor plan for given adjacency constraints, we can opt for the maximal rectangular floor plan because it covers all the adjacency requirements and the extra connections can be managed by considering sound proof walls or by better available techniques. If the number of rooms are not large, then a maximal rectangular floor plan can be transformed into a rectangular floor plan using Algorithm 2.

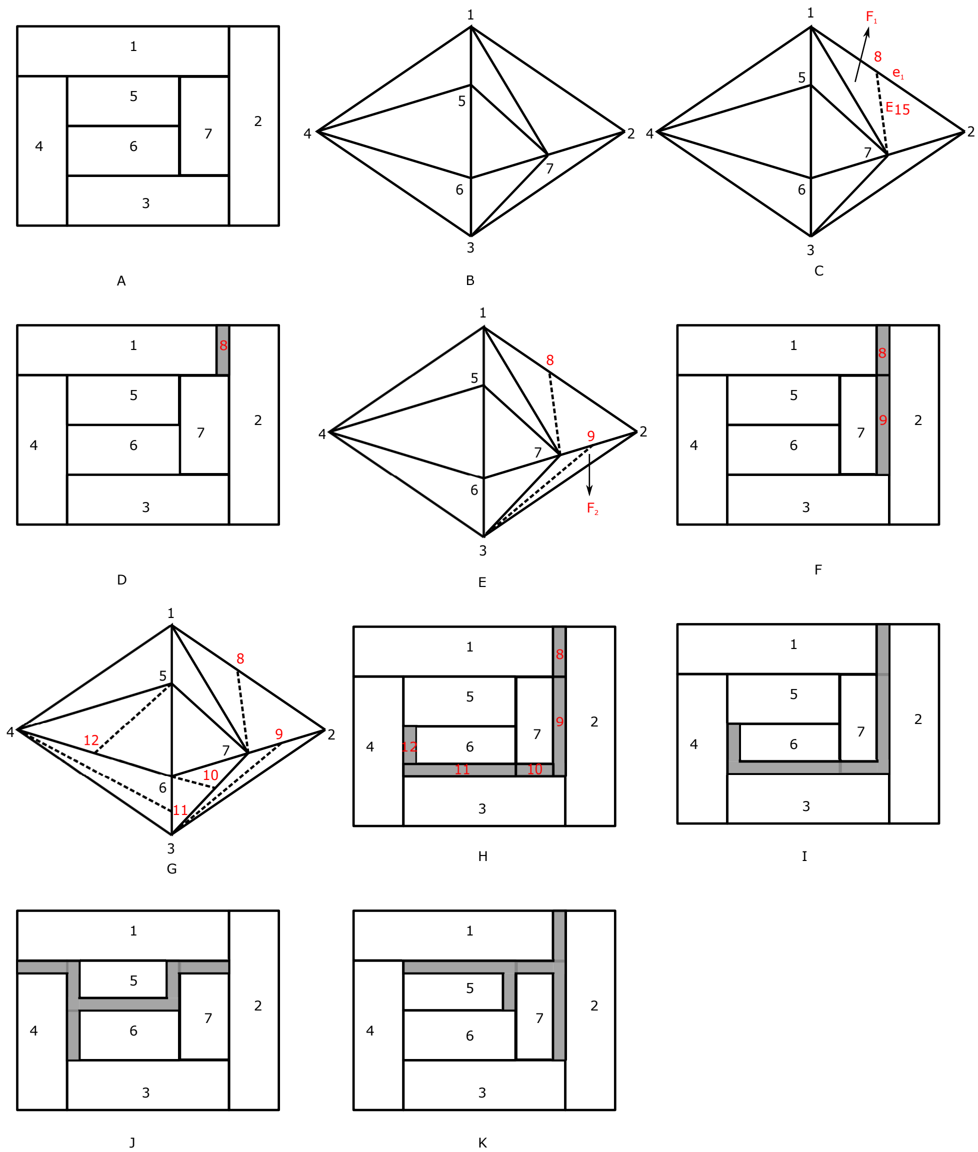

Fig. 11. Inserting circulation spaces within a maximal rectangular floor plan

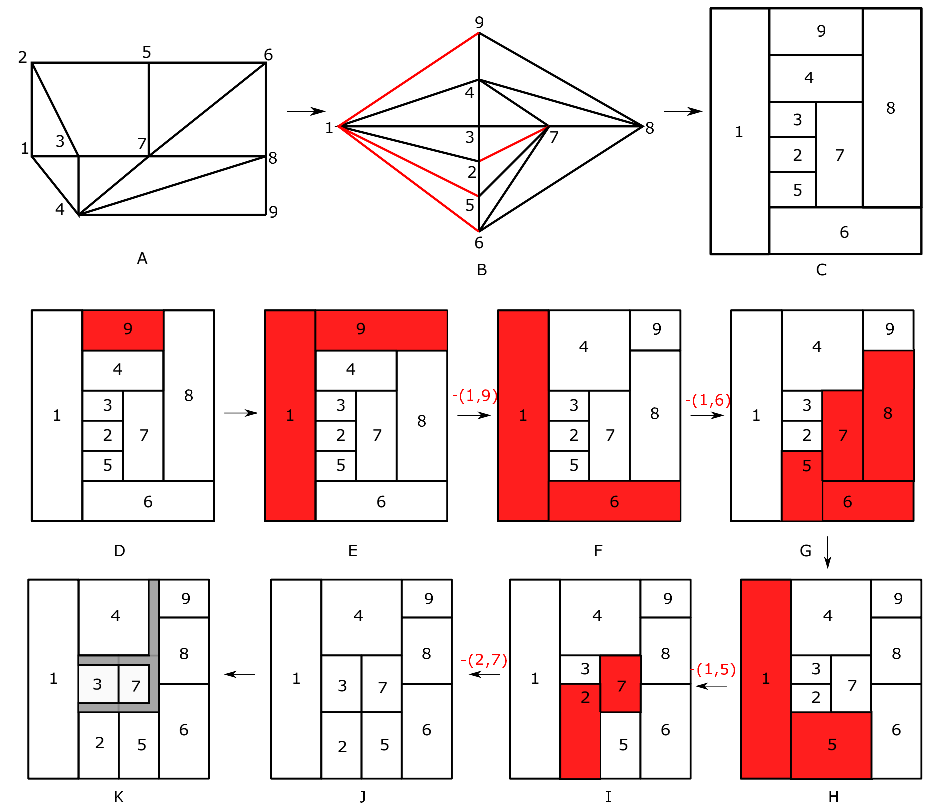

Fig. 12. Constructing a rectangular floor plan for the given adjacency graph and inserting circulations inside the obtained rectangular floor plan

– Maximal rectangular floor plans are very crucial for checking the existence of a rectangular graph for given adjacency requirements. If the given adjacency graph is not a sub-graph of any of the maximal rectangular floor plans, then there does not exist a rectangular floor plan for given graph otherwise it may or may not exist. To further check the existence of a rectangular floor plan, we start deleting the extra connections one by one while maintaining rectangularity and other adjacencies. If it is possible to preserve the rectangularity and adjacencies during the process of deletion of extra connections, we will have a rectangular floor plan for the given graph, otherwise some rooms need to be rectilinear that will give us an orthogonal floor plan for the given graph.
– It may happen sometimes that there does not exist a floor plan for the given adjacency requirements, i.e, the adjacency graph is non-planar. In this case, how do we know if it does not exist, i.e., we may keep trying to generate a floor plan which is not possible. Using Algorithm 2, we are the first one to check the planarity of given graph and to construct a floor plan corresponding to the non-planar adjacency graph.
– From the architectural point, circulations play a very important role in a building. Hence, we propose an algorithm (see Algorithm 3) for inserting spanning circulations inside a maximal rectangular floor plan. Here, we choose to enumerate spanning circulation so that every room is reachable to every other room through circulations.

Using this paper, we are the first one to introduce a mathematical procedure for constructing a floor plan corresponding to any given adjacency requirements. Once we have a floor plan, we present an algorithm for inserting circulations inside the obtained floor plan. Here, the idea is to provide a feasible floor plan along with circulations corresponding to any adjacency requirements. As a future work, we are planning to introduce dimensions to the constructed floor plan. Our larger aim is to build a generic system for the automated generation of architectural floor plans, i.e., we aim to provide architects/designers with design aids that can always generate good feasible solutions which can be further improved and adjusted by the architects.

Acknowledgment

The work presented in this paper is a part of the research project Mathematics-aided Architectural Design Layouts (File Number: ECR/2017/000356) funded by the Science and Engineering Research Board, India.

References

1. Levin PH (1964) Use of graphs to decide the optimum layout of buildings. *The Architects' Journal* 140(15): 809 − 817.
2. Steadman JP (1973) Graph theoretic representation of architectural arrangement. *Architectural Research and Teaching* 2/3: 161 − 172.

3. Lynes JA (1977) Windows and floor plans. *Environment and Planning B* 4: 51 − 55.
4. Baybars I and Eastman CM (1980) Enumerating architectural arrangements by generating their underlying graphs. *Environment and Planning B* 7: 289 − 310.
5. Baybars I (1982) The generation of floor plans with circulation spaces. *Environment and Planning B* 9: 445 − 456.
6. Rinsma I (1987) Nonexistence of a certain rectangular floorplan with specified areas and adjacency. *Environment and Planning B: Planning and Design* 14: 163 − 166.
7. Rinsma I (1988) Rectangular and orthogonal floorplans with required room areas and tree adjacency. *Environment and Planning B: Planning and Design* 15: 111 − 118.
8. Recuero A, Río O and Alvarez M (2000) Heuristic method to check the realisability of a graph into a rectangular plan. *Advances in Engineering Software* 31: 223 − 231.
9. Marson F and Musse SR (2010) Automatic Real-Time Generation of Floor Plans Based on Squarified Treemaps Algorithm. *International Journal of Computer Games Technology* 2010: 1 − 10.
10. Jokar MRA and Sangchooli AS (2011) Constructing a block layout by face area. *The International Journal of Advanced Manufacturing Technology* 54: 801 − 809.
11. Zhang H and Sadasivam S (2011) Improved floor-planning of graphs via adjacency-preserving transformations. *Journal of Combinatorial Optimization* 22: 726 − 746.
12. Regateiro F, Bento J and Dias J (2012) Floor plan design using block algebra and constraint satisfaction. *Advanced Engineering Informatics* 26: 361 − 382.
13. Shekhawat K (2014) Algorithm for constructing an optimally connected rectangular floor plan. *Frontiers of Architectural Research* 3(3): 324 − 330.
14. Shekhawat K (2015) Computer-aided architectural designs and associated covariants. *Journal of Building Engineering* 3: 127 − 134.
15. Shekhawat K and Duarte JP (2018) Introduction to generic rectangular floor plans. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 32(3): 331 − 350.
16. Wang X-Y, Yang Y and Zhang K (2018) Customization and generation of floor plans based on graph transformations. *Automation in Construction* 94: 405 − 416.
17. Shekhawat K (2018) Enumerating generic rectangular floor plans. *Automation in Construction* 92: 151 − 165.
18. Krishnamurti R and Roe P H O'N (1979) On the generation and enumeration of tessellation designs. *Environment and Planning B* 6: 191 − 260.
19. Mitchell WJ, Steadman JP and Liggett RS (1976) Synthesis and optimization of small rectangular floor plans. *Environment and Planning B* 3(1): 37 − 70.
20. Bloch CJ (1979) Catalogue of small rectangular plans. *Environment and Planning B* 6(2): 155 − 190.
21. Lee J, Han WS et al. (2012) An in-depth comparison of subgraph isomorphism algorithms in graph databases. *Proceedings of the VLDB Endowment* 6(2): 133 − 144.