A GRAPH THEORETIC APPROACH FOR THE AUTOMATED GENERATION OF DIMENSIONED FLOORPLANS

KRISHNENDRA SHEKHAWAT
[1]*Department of Mathematics, BITS Pilani, Pilani Campus, India-333031*
[1]*krishnendra.shekhawat@pilani.bits-pilani.ac.in*

Abstract. The automated generation of architectural layouts is an intensively studied research area where the aim is to generate a variety of (initial) layouts for the given constraints which can be further modified by designers and architects. From a mathematical perspective, one of the well-known constraints is given in the form of an adjacency graph which represents the adjacency relations of the given rooms and problem is to generate multiple layouts satisfying the adjacency relations. In the literature, the adjacency graph is usually taken as a bi-connected planar triangular graph. In this paper, we present the results of a prototype GPLAN that generates multiple dimensioned layouts for any given planar graph. The larger aim of this work is to develop software that can produce a variety of architecturally acceptable floorplans corresponding to the given constraints.

1. Introduction

From 1970s, a good amount of work has been done in the domain of computer-aided architectural design, where the prime focus is to automatically generate floor-plan layouts, so that these layouts are regarded as preliminary layouts by architects/designers and can be further modified and adjusted by them (Steadman 1973, Mitchell et al. 1976). If we restrict ourselves to the generation of floorplans corresponding to a given adjacency graph, a few of the relevant work in this direction is as follows:

The automated generation of architectural layouts using graph theory began with the generation of rectangular floorplans (RFP). It was first proposed by Levin 1964 in the early 1960s. Then in 1970s, many researchers proposed graph-theoretical algorithms for the enumeration or construction of rectangular layouts (Lynes 1977, Gilleard 1978, Baybars and Eastman 1980). During 1980s, Roth et al. 1982, Rinsma 1987 and Rinsma 1988 developed efficient graph algorithms for the construction of dimensionless and dimensioned RFPs with some restrictions on the input graph.Then in early 1990s, researchers realized that there are graphs for which RFPs do not exist (Rinsma et al. 1990, Yeap and Sarrafzadeh 1993). Liao et al. 2003 gave a linear time algorithm for constructing a dimensionless orthogonal floorplan (OFP) for any planar triangulated graph (PTG). Jokar and Sangchooli 2011 used the concept of face area of a graph for the construction of dimensionless OFPs for given PTGs. Eppstein et al. 2012 gave an

PROJECTIONS, Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2021, Volume 1, 141-150.

algorithm for finding an area universal RFP for the given adjacency requirements whenever such layout exists. Wang et al. 2018 gave the automated regeneration of well-known existing dimensionless RFPs while considering underlying adjacency graph of the existing floorplan. In the same year, Shekhawat 2018 enumerated all possible maximal RFPs without considering dimensions of the rooms. Recently, Upasani et al. 2020 developed a prototype for generating dimensioned RFPs for any drawn rectangular arrangement, satisfying adjacency, size and symmetric requirements. In the same year, Wang and Zhang 2020 extended GADG (Wang et al. 2018) for generating dimensioned OFPs corresponding to user-specified design requirements.

It can be observed from the literature that the construction of floorplans is restricted to bi-connected planar triangulated graphs only while for many architectural layouts the underlying graph is either 1-connected or non-triangulated. In this work, we present the results of prototype GPLAN that generates floorplans for any given planar connected graph. The obtained floorplans are categorized into two forms: the one having the rectangular boundary and other ones having the non-rectangular boundary.

2. Terminologies

In this section, we present the definitions related to adjacency graph and floorplans, which will be used further in the next Section.

A floor plan is a polygon, the plan boundary divided by straight lines into component polygons called rooms. If the boundary as well as rooms of a floorplan are rectangular it is called rectangular floorplan (RFP). If the boundary is rectangular while some rooms are rectilinear, it is called orthogonal floorplan (OFP). In a non-rectangular floorplan (NRFP), boundary is rectilinear while the rooms can be rectangular or rectilinear. For an illustration refer to Figure 1, where RFP, OFP and NRFP are shown.

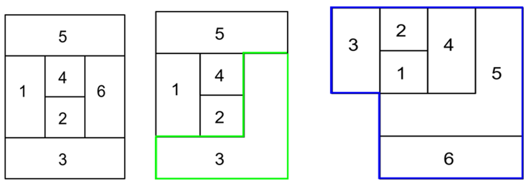


Figure 1. Floorplans Characterization.

Corresponding to a floorplan, there always associates a graph known as dual graph which can be constructed by taking each room as a vertex and drawing an edge between two vertices if and only if corresponding rooms share a wall or a part of a wall (see Figure 2).

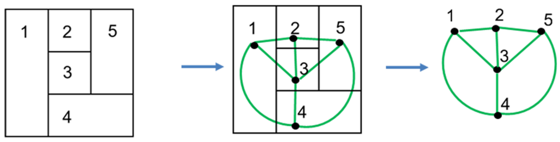


Figure 2. Constructing dual graph of a floorplan.

Moving from a floorplan to a graph is trivial but constructing a floorplan for a given graph is not easy and therefore, we will discuss it in detail in the next section. For example, consider the graph in Figure 3 for which constructing a floorplan is not trivial by hand and an algorithm is required to deal with such graphs having a large number of vertices.

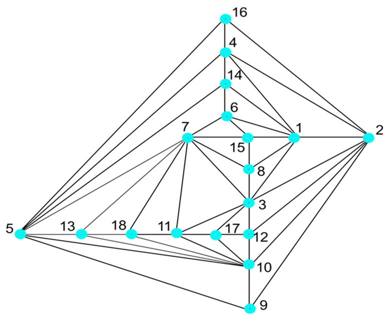


Figure 3. An adjacency graph for which constructing a floorplan manually is not easy.

A connected graph is said to be planar if it can be drawn in plane with edge crossings. A plane graph is a planar graph with an embedding that divides the plane into connected components called faces/regions. The unbounded region is called external face. Except for the external face, all other faces are internal faces. A planar triangulated graph (PTG) is a planar graph having all of its faces triangular. For example, the graph in Figure 4a is not planar while in Figure 4b is a PTG.

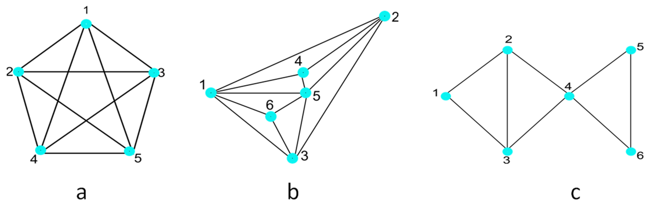


Figure 4. Illustrating planarity and connectivity of a graph.

A graph G is said to be 1-connected if it has a cut-vertex where a vertex v is a cut-vertex if its removal disconnects the graph. A graph G is said to be bi-connected if it has no cut-vertex. For example, the graph in Figure 4c is 1-connected (vertex 4 is the cut-vertex) while the graph in Figure 4b is bi-connected having no cut-vertices.

Since a floorplan itself can be regarded as a planar connected graph, therefore from here onwards an adjacency graph refers to a planar connected graph.

3. Methodology

In this section, we present the construction of dimensioned floorplans for any given adjacency graph G that are produced using the prototype GPLAN. It is important to note here that G mainly represents the adjacency relations among the rooms. At the same time, G also serves the following two purposes which are not directly visible:

- The geometry of the rooms: We are not asking for the room geometries as an input but they are derived based on the adjacency relations, i.e. in order to satisfy the given adjacencies, the rooms can be rectangular or rectilinear. For example, for the graph in Figure 5, at least one of the room in the corresponding floorplan needs to be rectilinear for satisfying the given adjacencies. In the floorplan in Figure 5, room 1 is of T-shape .
- Adjacency with the exterior: While drawing an adjacency graph, the user has a choice to keep any of the rooms in the interior or at the exterior. For example in Figure 5, rooms 1, 2 and 3 are interior rooms while 4, 5 and 6 are exterior rooms.

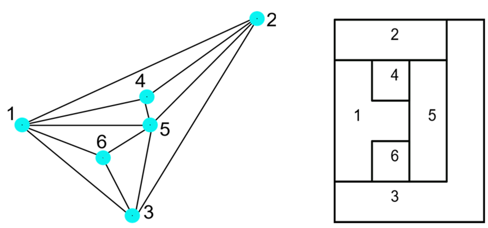


Figure 5. Illustrating geometry of the rooms and their relation with the exterior.

To proceed further, we consider the following cases based on the nature of the given graph G:

- 1. If the given graph G is planar, bi-connected and triangulated

In this case, it is always possible to construct a floorplan with rectangular boundary but the rooms can be rectangular or rectilinear. For example, GPLAN construct the required floorplans in Figure 6 where there exists a RFP for the first graph but RFP does not exist for the second graph and therefore OFP has been illustrated. Also in the RFP, all rooms are adjacent to the exterior while in the OFP, only room 5 is at the interior.

For the construction of a RFP, a linear time algorithm has been developed which is core of all the results of the paper, i.e., modifying this algorithm, floorplans in the consequent steps have been obtained.

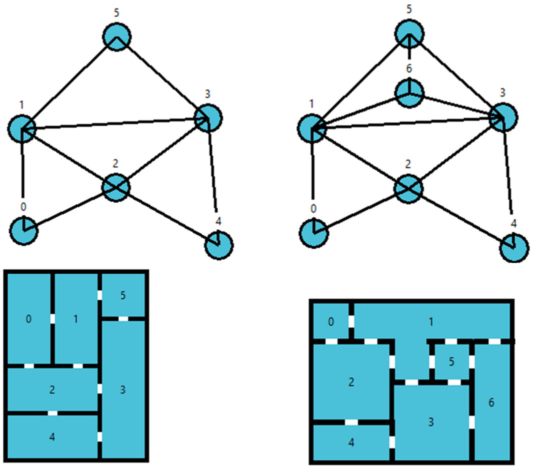


Figure 6. RFP and OFP corresponding to given graphs.

As mentioned in the Introduction, most of the work is restricted to planar bi-connected and triangulated graphs and it is covered in Step 1. Now in the coming steps, we address the construction of floorplans where the input graph is neither bi-connected nor triangulated.

- 2. If G is a planar bi-connected graph but not triangulated

If the graph is not triangulated, it means there exists at least a cycle of length greater 3 in the graph. Architecturally it is interesting to note that a cycle of length greater than 3 in a graph represents a corridor, i.e., to satisfy given adjacencies, corridors need to be introduced in the corresponding floorplan (see Figure 7C).

If the graph is not triangulated, the first step is to triangulate it by adding new edges (the graph in Figure 7A is not triangulated which is further triangulated in Figure 7B by adding 3 new edges). Now because of the new edges, we have two floorplan representations as follows:

- Floorplans with corridors: After triangulation, the floorplan F can be obtained as in Step 1. Now in the obtained floorplan F, if the additional edges or connections are removed, then we get a floorplan with corridors corresponding to the given graph as shown in Figure 7C. Clearly, in Figure 7A, there are two cycles of length greater 3 which leads to the generation of two corridors as shown in Figure 7C.
- Floorplan with and without doors: Also in the obtained floorplan F, the original adjacencies can be represented by doors while the new added adjacencies can be represented by the walls as shown in Figure 7D. For example, since room 1 and room 3 are not adjacent in Figure 7A, there is no door between them in Figure 7D while there is a door between rooms 0 and 3. This representation can be used when corridors are not required and floorplans need to be more compact and flexible.

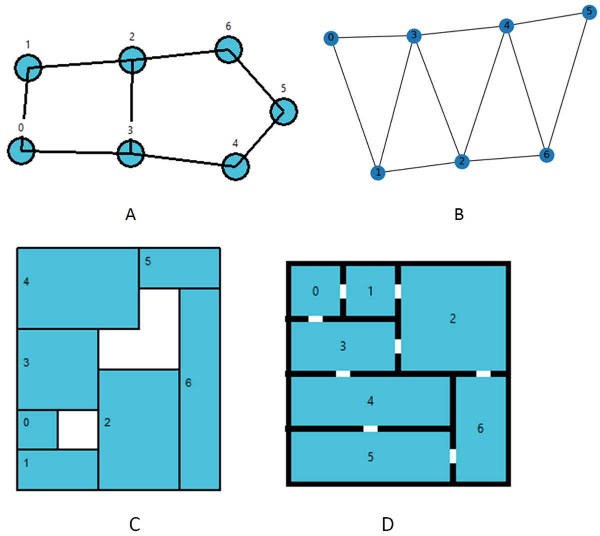


Figure 7. Corresponding to a given non-triangulated graph, a floorplan with corridors and a floorplan with doors.

- 3. If G is a planar 1-connected triangulated graph

In this case, we first add extra edges to make the graph G bi-connected and obtain a floorplan F using Step 1. For example, for the graph in Figure 8A, which is 1-connected, corresponding bi-connected graph is shown in Figure 8B. Now again added connections can be removed from F to have a floorplan with corridors or added connections can be represented by walls in F as done in Step 2 (refer to Figure 8C and 8D respectively).

The 1-connected graphs are helpful in representing different blocks of a building. For example, in Figure 9B, rooms 0, 1, 2 and 3 form a block which is adjacent to other block of building via room 3. Also, using 1-connected graphs, it is easy to represent rooms which requires isolation or a minimum connectivity. For example, in Figure 10C, room 5 is only adjacent to room 2.

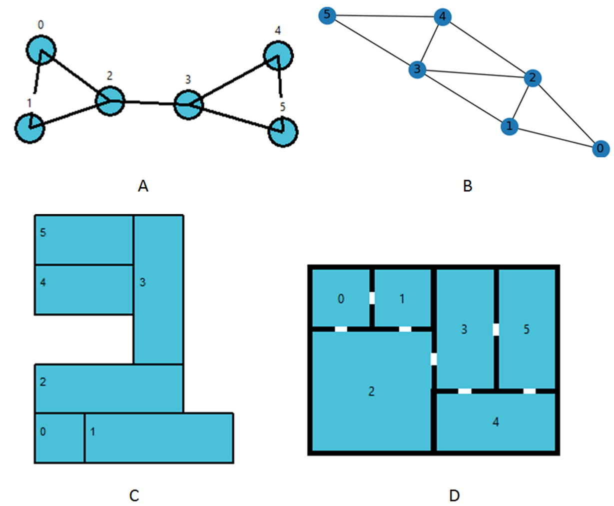


Figure 8. 1-connected graph and corresponding floorplans.

- 4. If G is a planar 1-connected graph and not triangulated

In this case, we first make G bi-connected and then triangulate it. Then using the concepts of Steps 2 and 3, we will have required floorplans. For example, refer to Figure 9A where the given graph is non-triangulated and 1-connected and floorplans corresponding to it are illustrated in Figures 9B and 9C.

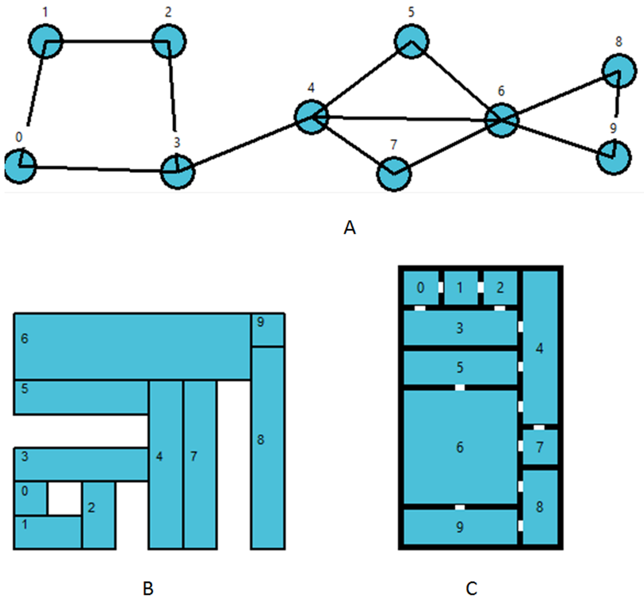
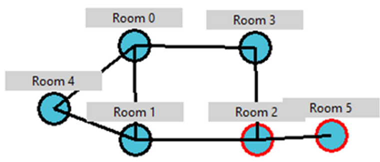


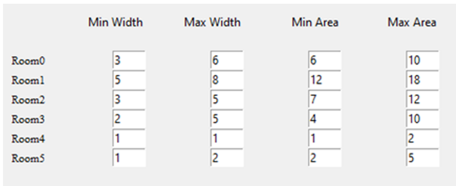
Figure 9. A 1-connected non-triangulated graph and corresponding floorplans.

- 5. Dimensioned floorplans

Once we obtain the required floorplans, the dimension of each room is taken as input in the form of minimum and maximum width along with minimum and maximum area. For example, refer to Figure 10B where dimensional constraints are illustrated. A dimensioned floorplan for the graph given in Figure 10A is shown in Figure 10C.

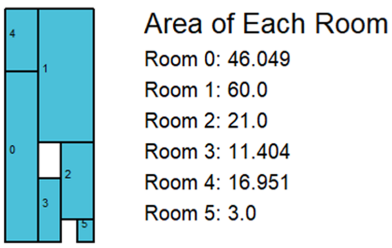


A



	Min Width	Max Width	Min Area	Max Area
Room0	3	6	6	10
Room1	5	8	12	18
Room2	3	5	7	12
Room3	2	5	4	10
Room4	1	1	1	2
Room5	1	2	2	5

B



C

Figure 10. Dimensioned floorplan for the given graph.

Remark. In Figures 7, 8 and 9, we have proposed two floorplan representation for the given adjacency graph. In both the representations, all adjacency relations are satisfied but in the second one, the boundary is always rectangular and empty spaces are not allowed (see Figures 7D, 8D and 9C) while in the first one, empty spaces as well as non-rectangular boundary are considered (see Figures 7C, 8C and 9B). Architecturally each one has its importance as discussed above but mathematically, they differ on the basis of compactness.

4. Conclusion and Future Work

In this paper we have presented the results of an operational software GPLAN where the purpose is to generate dimensioned floorplans for a given graph. It can be found in the literature that the most of the work is restricted to planar bi-connected triangulated graph, i.e., the 1-connected graph and non-triangulated graphs were neglected which are quite crucial for representing some interesting features of architectural layouts. For example, non-triangulated graphs directly shows the requirement of corridors while 1-connected graph represents different blocks that are connected through a cut-vertex.

Furthermore, GPLAN can also be regarded as a tool for regenerating existing floorplans by considering the dual graph of existing floorplan. For example, in Figure 11, Villa Trissino has been regenerated using GPLAN.

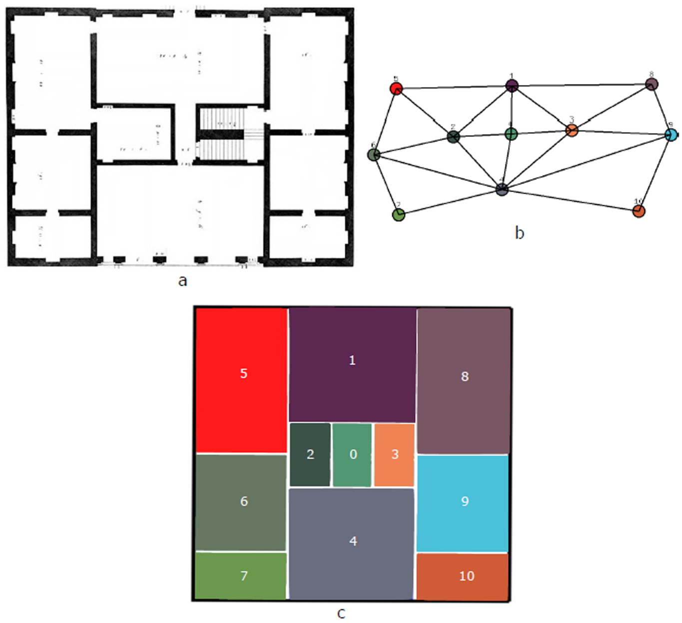


Figure 11. Regenerating Villa Trissino.

At this stage, many important components are not covered by GPLAN, few of them are circulations, boundary constraints, daylight constraints etc., which we are planning to cover in near future. Also, we agree that the proposed designs at this stage may not be architecturally acceptable and therefore we are thinking of introducing data-driven approaches (Hu et al. 2020) to GPLAN so that GPLAN provides multiple initial layouts to designers which can be further modified.

References

Baybars, I. and Eastman, C.: 1980, Enumerating architectural arrangements by generating their underlying graphs, *Environment and Planning B*, **7**, 289-310.

Eppstein, D., Mumford, E., Speckmann, B. and Verbeek, K.: 2012, Area-universal and constrained rectangular layouts, *SIAM Journal on Computing*, **41**(3), 537-564.

Gilleard, J.: 1978, Layout - a hierarchical computer model for the production of architecturalfloor plans, *Environment and Planning B*, **5**, 233-241.

Hu, R., Huang, Z., Tang, Y., Matias, O., Kaick, V., Zhang, H. and Huang, H.: 2020, Graph2Plan:

learning floorplan generation from layout graphs, *ACM Transactions on Graphics*, **39**(4), 1-14.

Jokar, M.R.A. and Sangchooli, A.S.: 2011, Constructing a block layout by face area, *The International Journal of Advanced Manufacturing Technology*, **54**(5-8), 801-809.

Levin, P.H.: 1964, Use of graphs to decide the optimum layout of buildings, *The Architects' Journal*, **7**, 809-815.

Liao, C.C., Lu, H.I. and Yen, H.C.: 2003, Compact floor-planning via orderly spanning trees, *Journal of Algorithms*, **48**(2), 441-451.

Lynes, J.A.: 1977, Windows and floor plans, *Environment and Planning B*, **4**, 51-55.

Mitchell, W.J., Steadman, J.P. and Liggett, R.S.: 1976, Synthesis and optimization of small rectangular floor plans, *Environment and Planning B: Planning and Design*, **3**(1), 37-70.

Rinsma, I.: 1987, Nonexistence of a certain rectangular floorplan withspecified areas and adjacency, *Environment and Planning B*, **14**, 163-166.

Rinsma, I.: 1988, Rectangular and orthogonal floorplans with required room areas and tree adjacency, *Environment and Planning B: Planning and Design*, **15**(1), 111-118.

Rinsma, I., Giffin, J. and Robinson, D.: 1990, Orthogonal floorplans from maximal planar graphs, *Environment and Planning B: Planning and Design*, **17**(1), 57-71.

Roth, J., Hashimshony, R. and Wachman, A.: 1982, Turning a graph into a rectangular floor plan, *Building and Environment*, **17**(3), 163-173.

Shekhawat, K.: 2018, Enumerating generic rectangular floor plans, *Automation in Construction*, **92**, 151-165.

Steadman, P.: 1973, Graph theoretic representation of architectural arrangement, *Architectural Research and Teaching*, **2(3)**, 161-172.

Upasani, N., Shekhawat, K. and Sachdeva, G.: 2020, Automated generation of dimensioned rectangular floorplans, *Automation in Construction*, **113**, 103149.

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.

Wang, X.Y. and Zhang, K.: 2020, Generating layout designs from high-level specifications, *Automation in Construction*, **119**, 1-12.

Yeap, K.H. and Sarrafzadeh, M.: 1993, Floor-planning by graph dualization: 2-concave rectilinear modules, *SIAM Journal on Computing*, **22**(3), 500-526.