Identifying the Design Feature That Causes Project Delay in DfMA
A Dominant Element Analysis Method for Project Scheduling

Cao, Jianpeng; Zhang, Hang; Pan, Bo; Soman, Ranjith; Savov, Anton; Hall, Daniel

Publication date
2024

Document Version
Final published version

Published in
Computing in Civil Engineering 2023

Citation (APA)
Cao, J., Zhang, H., Pan, B., Soman, R., Savov, A., & Hall, D. (2024). Identifying the Design Feature That Causes Project Delay in DfMA: A Dominant Element Analysis Method for Project Scheduling. In Y. Turkan, J. Louis, F. Leite, & S. Ergan (Eds.), *Computing in Civil Engineering 2023: Visualization, Information Modeling, and Simulation - Selected Papers from the ASCE International Conference on Computing in Civil Engineering 2023* (pp. 163-170). American Society of Civil Engineers (ASCE). https://ascelibrary.org/doi/10.1061/9780784485231.020

Identifying the Design Feature That Causes Project Delay in DfMA: A Dominant Element Analysis Method for Project Scheduling

Jianpeng Cao[1]; Hang Zhang[2]; Bo Pan[3]; Ranjith Soman[4]; Anton Savov[5]; and Daniel Hall[6]

[1]Dept. of Civil and Environmental Engineering, Institute for Construction and Infrastructure Management, ETH Zürich. Email: cao@ibi.baug.ethz.ch
[2]Dept. of Architecture, Institute of Technology in Architecture, ETH Zürich. Email: hangzhang@ethz.ch
[3]Dept. of Architecture, Institute of Technology in Architecture, ETH Zürich. Email: bopanb@student.ethz.ch
[4]Dept. of Civil and Environmental Engineering, Institute for Construction and Infrastructure Management, ETH Zürich. Email: soman@ibi.baug.ethz.ch
[5]Dept. of Architecture, Institute of Technology in Architecture, ETH Zürich. Email: savov@arch.ethz.ch
[6]Assistant Professor, Faculty of Architecture and the Built Environment, TU Delft, Netherlands. Email: D.M.Hall@tudelft.nl

ABSTRACT

Design for manufacturing and assembly (DfMA) is an engineering methodology which aims to increase ease of manufacture and efficiency of assembly by considering manufacturing and assembly constraints in the design process. However, current DfMA approaches in the construction sector are not automated enough to identify the design features that may cause project delay in real time. This leads to longer design cycle. Also, current scheduling algorithms rely on human intervention to generate activity network from a design output. Addressing these inefficiencies, we propose an interpretative machining learning model to predict the construction duration given a design output. More importantly, the same model identifies the design features that may cause the most delay in the project. The model is trained on a residential design dataset with various features, such as layout, geometry, and element typology. The output of the model is the project duration and an importance map, indicating the influence each feature of the given design has on the total project duration. The results from this model can considerably reduce the design cycle by supporting architects to create fabrication and assembly aware design even when they have little knowledge of production and assembly processes. This model will contribute to a novel computational approach for DfMA.

INTRODUCTION

The architecture, engineering and construction (AEC) sector must respond to an ongoing global housing crisis. By 2025, a third of the world's urban dwellers—1.6 billion people—could struggle to secure adequate housing (Woetzel, 2014). In recent years, Europe has faced an exceptionally high demand for urban housing. This demand is fueled primarily by urbanization, increased immigration, and demographic trends towards one-person-households.

To meet this demand, the European housing industry needs to develop high-end, customized solutions. However, it is questionable if the current system for the design and construction of housing is responsive enough to rapidly deliver customized housing without compromising architectural quality. In current scenario, the architects must propose a completed design with details for the construction duration estimation. The feedback from contractor is not real-time but takes several days, also it is unable to point out the main reason or main feature causing a long duration, which results in a long-time, iterated “design modification – calculation” process that stultify the efforts of architects.

New technology can help. Design for manufacturing and assembly (DfMA) is a new engineering methodology which aims to increase ease of manufacture and efficiency of assembly by considering manufacturing and assembly constraints in the design process (Lu et al., 2021). However, current DfMA approaches in the construction sector are not automated enough to identify the design features that may cause project delay in real-time.

Longer construction duration caused mainly by large amount of complicated building components, that’s the reason why a detailed design is required for estimation nowadays. However, the amount and the complexity of building component is foreshadowed in the initial floorplan design, as the physical configuration of walls is dependent on the function and spatial relationship of rooms.

The novel contribution of this research will be an interpretative machining learning model which can give the architects an immediate duration estimation even during the early design stage without development in details, and an importance map indicating which part or which elements in the floorplan contributes most to a long duration. In order to achieve this, three research questions were designed:

1. How could early-stage floorplan be created using a generative model?
2. How could fabrication duration be calculated based on domain knowledge by design synthesis?
3. How could the duration be predicted, and the most important design features be identified?

DATA ACQUISITION

We use a large-scale benchmarking dataset, RPLAN (Wu et al., 2019). RPLAN is an image dataset sampled from real-world residential layouts in Asia, that consists over 80000 floorplans. As all the designs are within a squared region of 18m × 18m, we fix the scale to retrieve the real-world value of room sizes from image. While RPLAN contains 13 types of rooms, we merge the same function type labels (e.g., master room and child room are all treated as bedroom).

Each of the floorplan layouts in the dataset is denoted as a combination of a site condition B = {F, M} and a layout graph G = {T, C, A, S, R}, where F∈R^2 is the centra location of front door, and similar to the previous works (Nauata et al., 2020; Shi et al., 2020), M∈R ^ (128×128) refers to the boundary image. The layout graph G is representing geometrical and categorical features of rooms, parameterized by the room types T, the center coordinates C of the room rectangular regions, the adjacency matrix A indicating rooms spatial relations, the corresponding room sizes S, and the rooms aspect ratio R. Length. Length is determined by the conference technical committee or proceedings editor. Total paper length includes all text, graphics, references, and appendixes.

GENERATIVE MODELS

While the layout planning is initialized by giving a specific site condition I, the properties of each layout features in G might varies (e.g., room types T containing 1 bath or 2 baths). The ideal generative model will be structured to learn P(G), the potential distributions of all the features in the layout. Our approach decodes these features procedurally and iteratively, that all the confirmed features will be conditional input to predict the next one. Structured as a joint probability distribution, the whole generation process will be formulated as Markov Chain (Figure 1), while the task P(G) is decomposed as:

$$\begin{gathered}\mathcal{P}(G) = \mathcal{P}(R|S,A,C,T,I)\mathcal{P}(S|A,C,T,I) \\ \mathcal{P}(A|C,T,I)\mathcal{P}(C|T,I)\mathcal{P}(T|I)\mathcal{P}(I)\end{gathered} \tag{1}$$

where P(I) is the given site condition, P(T│I), P(C│T,I), P(A│C,T,I), P(S│A,C,T,I), and P(R│S,A,C,T,I) respectively denote the step-wise forecasting of types T, centers C, adjacency A, size S, and ratio R.

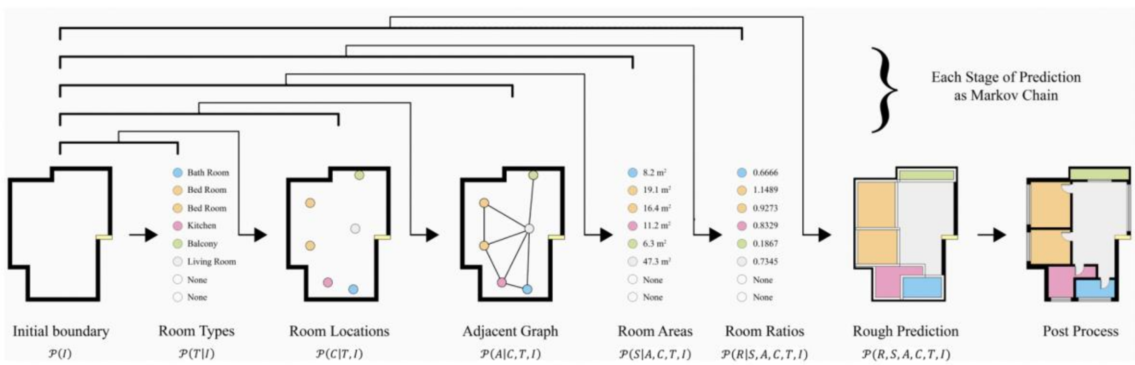

Figure 1. Layout generation process as Markov Chain

DESIGN SYNTHESIS

Design synthesis is an activity to develop the physical configuration capable of performing required functions. The physical configuration defines how building elements are fabricated and assembled, such as multi-layers of wall elements. The objective of design synthesis on the created layout is to specify the element physical configuration so that the construction duration could be estimated. In this study, we conduct design synthesis on wall elements for a given layout. The wall architecture could be defined by from two perspectives, functional layers and service layers.

First, the functional layers could be determined by the adjacent rooms (as Figure 2). Taking a cue from the industrial practice, the walls, or prefabricated panels, must be divided into different categories with different layer configurations after the floor plan layout has been generated. The criteria for categorizing are primarily based on the demands for thermal insulation, waterproofing, and soundproofing. These specifications depend on how the rooms on either side of the panel are used. Thermal insulation and a waterproof coating are necessary for exterior walls that separate the indoor and outdoor environments. A waterproof is also necessary for interior walls separating "dry" rooms like the kitchen, bathroom, and other. The purpose of the rooms on the other side will determine whether a soundproof layer is necessary for noise-sensitive rooms.

In the industrial process, the panels are put together from various functional layers, each of which requires a specific amount of time for production. As a result, the panels with different layer configurations require different amounts of time for production, which can be characterized as the sum of all the time units used by each layer. To simplify the duration calculation, the time unit consumed by each layer is set as 0.5. The core structure panel consumes 1 unit time, each time a layer is added, the time unit will be increased by 0.5. Figure 3 indicates the production time for each type of wall according to adjacent room requirements.

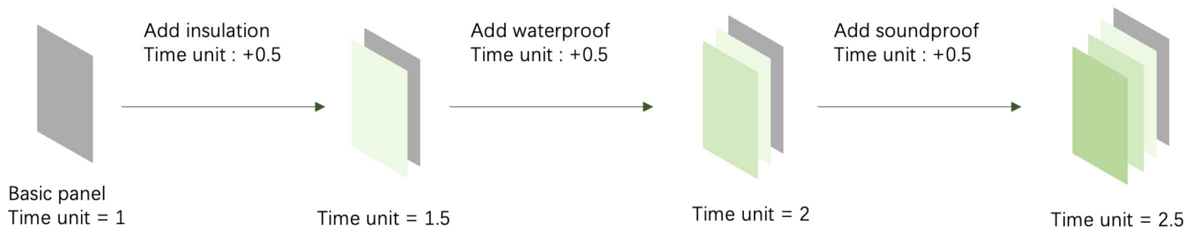

Figure 2. Physical architecture of walls and corresponding task duration

	0_None	1_Entray	2_Bathroom	3_Closet	4_Bedroom	5_Kitchen	6_Dining	7_Balcony	8_Living room
0_None	1	2	2	1.5	2.5	2	2	1	2
1_Entray		1	1.5	1	2.5	1.5	1.5	1.5	1.5
2_Bathroom			2	2	2.5	2	2	2	2
3_Closet				1	1	1	1	2	1
4_Bedroom					1.5	2	2	2.5	2
5_Kitchen						1	1	2	1
6_Dining							1	2	1
7_Balcony								1	2
8_Living room									1

Figure 3. Wall production time under adjacent room requirements

Second, the service layer is installed inside the wall for a ducting system. For a given layout plan, a starting point of the ducting system, as well as rooms with heating requirements are predefined. In this study, we set a wall from the kitchen as the starting point, and the living room as the heating room. Given above assumptions, a ducting layout could be generated using shortest path algorithm to minimize the ducting length and reduce the number of walls with a service layer (as Figure 4). The algorithm consists of following procedures:

1. Find the path from the starting point to each wall that belong to the heating room.
2. Calculate the ducting length by summing the length of walls on each path.
3. Find the path with the shortest length and set the walls on that path with service property.

Finally, the walls with a service layer could be identified and the time for installing service layer is added to the production time.

FABRICATION-AWARE MODEL

Given a generated layout, we consider the problem as training a regression model to predict the production duration of the layout, and training an explanation model to generate an importance index map for each feature. A higher index indicates that the relevant feature has a higher influence to the duration.

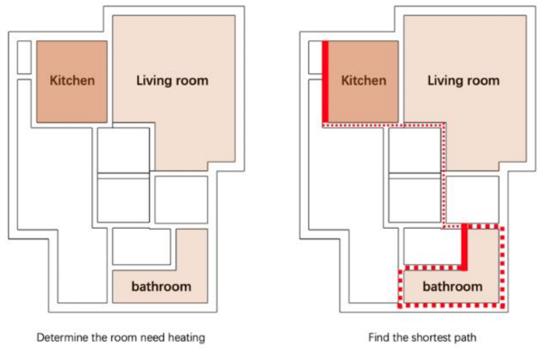

Figure 4. An approach to determine the ducting layout for a given floorplan

- **Regression model**

Graph neural networks (GNN) are used for the regression work. We follow the pipeline (Zhou et al., 2020), finding graph structure, specifying graph type and scale, designing loss function, selecting computational modules. We consider two graph structures, a room graph and a wall graph, both of which represent a same layout. The graphs are undirected and homogeneous. In a room graph, nodes represent rooms and edges represent adjacent relationship between rooms. In a wall graph, nodes represent walls while edges represent wall to wall connections. Specifically, continuous straight walls are divided at each intersection. Figure 5 gives an example of a room graph and a wall graph for a layout. The room types or wall types are assigned to node features via one-hot encoding. There are 8 room types, including outside, entry, bathroom, closet, bedroom, kitchen, dining, balcony and living room. There are 45 wall types, which are determined by adjacent rooms, such as bedroom-kitchen wall.

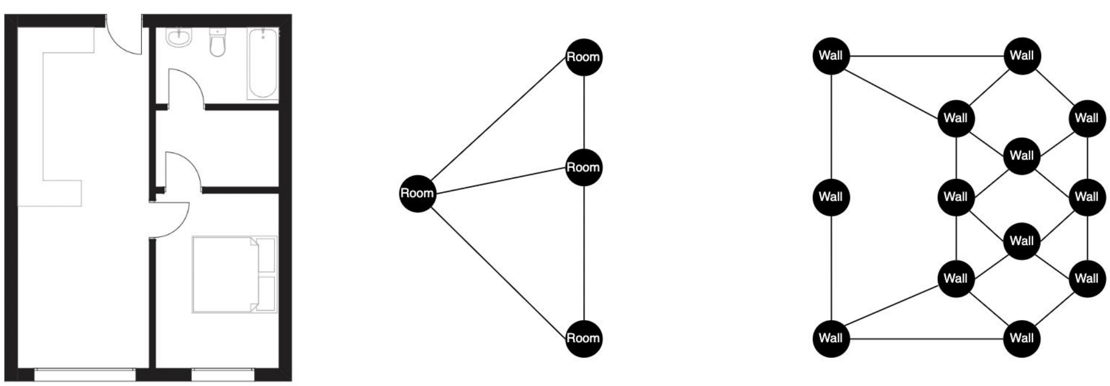

Figure 5. A layout (left) and a corresponding room graph (middle) and wall graph (right)

The duration prediction is defined as a graph regression task. The mean absolute error (MAE) is selected as the loss function. The proposed architecture is shown in Figure 6, where a number of Graph Convolutional Networks (GCN) (Kipf & Welling, 2017) are used to get the graph embedding, followed by a multilayer perceptron (MLP) neural network for the regression task. Borrowing the idea of mini-batch training from common deep learning practice, a batch of multiple graphs are built together and send for one training iteration.

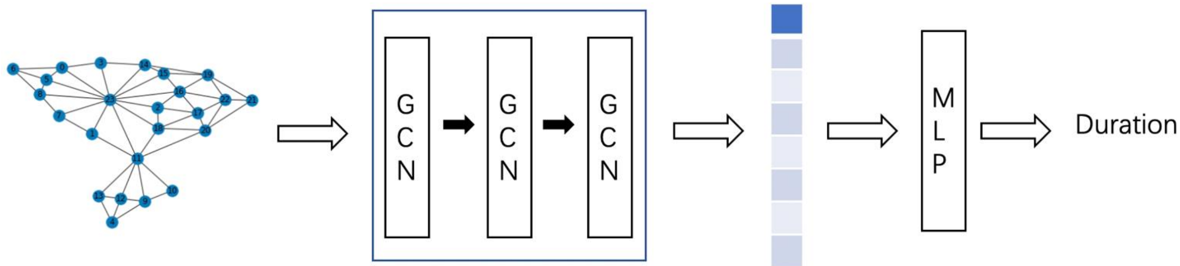

Figure 6. GNN-based regression model for the duration prediction

- **Explanation model**

The goal of explanation model is to identify compact subgraph structures and small subsets of node features that play a critical role in GNN-based graph regression. We adapted a GNN Explainer (Ying et al., 2019) which is originally designed for classification task and has not been validated in the regression task. GNN Explainer aims to find the feature importance by a node mask and an edge mask. A new prediction could be made after applying the feature mask on the node features and the edge mask on the graph. A mutual information that measures the probability change of prediction is set as the objective to be maximized. The large change of the prediction distribution indicates that the absence of the feature or the edge is a good counterfactual explanation.

$$\max_{G,F,M} MI\ \left(Y, (G, F, M)\right) = H(Y) - H(Y|G = G_s^M, X = X_s^F) \quad (2)$$

Where G_s is the subgraph and X_s is the associated features. $G_s = G \odot M$, M is the edge mask that has the same size of the adjacency matrix of the graph, $X_s^F = X_s \odot F$, F is the feature mask to be learned. $\odot$ denotes the element-wise multiplication. To make the importance index within [0,1], a sigmoid function is applied to the mask.

RESUTLS

We investigate questions: does the GNN-based model obtain a good prediction result? How does the explanation generated by GNN Explainer compare with the ground-truth knowledge.

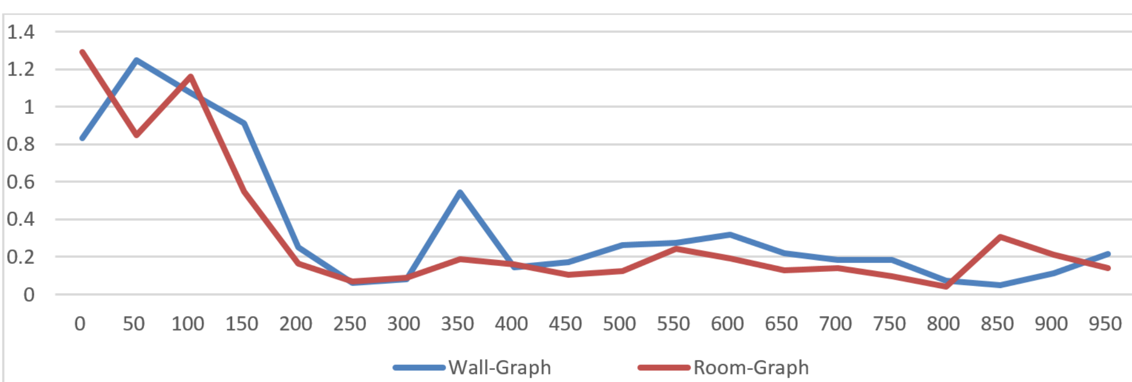

Figure 7. Experiment results

From the Figure 7, we can see the regression model using wall graphs and room graphs performs a relatively good result with test error in [15% - 20%]. From the Figure 8, a layout is used as an example to test the explanation. The edge thickness indicates the importance index. There is a small difference between room-level explanation and wall-level explanation. The room-level explanation covers a larger portion of the layout. Considering that room-level has a coarse granularity, any wall components within that room could contribute to the room importance.

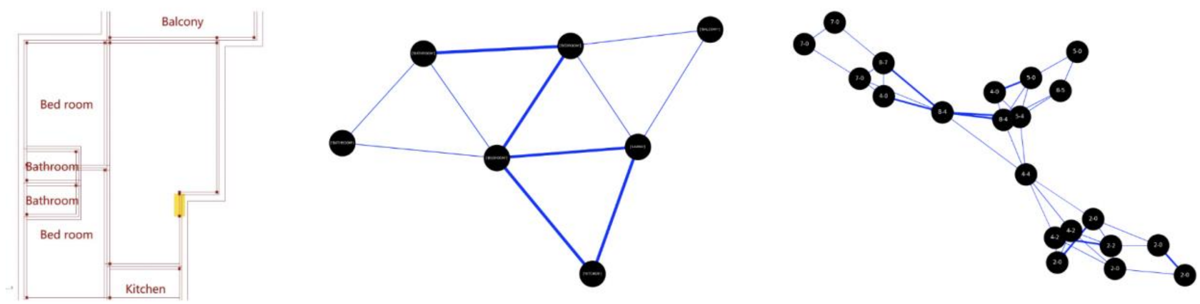

Figure 8. A layout example (left) and explanation by room graph (middle) and wall graph (right)

CONCLUSION

This research proposes an interpretative machining learning model that can provide architects with an instant duration estimation even during the early design stage without development in details and identify the dominant feature in layout causing the delay. This can help to reduce the recurring cycles between contractors and architects, therefore shorten the time needed for the completion of whole project.

The methodology is developed in 3 steps, Firstly, each of the conceptual design produced by generative model will be represented in two ways, the one is a “room graph” and the other “wall graph”, which both contain their attributes and location information. Secondly, the duration of project is estimated by summing up the production time of each wall. Thirdly, given a generated layout, a regression model is trained to predict the production time, and at the same time an explanation model is trained to generate an importance index map for each feature.

Admittedly, the model we are presenting is suffered from the following limitations. Firstly, the size of sample set is still restricted, causing the suboptimal performance of the model. Secondly, the attributes retrieved from the floorplan for prediction and explanation are not comprehensive enough, for example the joints of panels, the connection method of adjacent rooms, whether there are openings on the panel haven’t been taken into consideration. Future research should be undertaken to overcome these challenges and fully exploit the potential of interpretative generative model.

REFERENCES

Kipf, T. N., and Welling, M. (2017). Semi-Supervised Classification with Graph Convolutional Networks. *International Conference on Learning Representations*, 1–14.

Lu, W., Tan, T., Xu, J., Wang, J., Chen, K., Gao, S., and Xue, F. (2021). Design for manufacture and assembly (DfMA) in construction: the old and the new. *Architectural Engineering and Design Management*, 17(1–2), 77–91.

Nauata, N., Chang, K.-H., Cheng, C.-Y., Mori, G., and Furukawa, Y. (2020). House-GAN: Relational Generative Adversarial Networks for Graph-constrained House Layout Generation. *European Conference on Computer Vision*, 162–177. http://arxiv.org/abs/2003.06988.

Shi, F., Soman, R. K., Han, J., and Whyte, J. K. (2020). Addressing adjacency constraints in rectangular floor plans using Monte-Carlo Tree Search. *Automation in Construction*, 115(April), 103187.

Woetzel, J. R. (2014). *A blueprint for addressing the global affordable housing challenge*. McKinsey Global Institute, October, 30.

Wu, W., Fu, X., Tang, R., Wang, Y., Qi, Y., and Liu, L. (2019). Data-driven Interior Plan Generation for Residential Buildings. *ACM Transactions on Graphics*, 38(6).

Ying, R., Bourgeois, D., You, J., Zitnik, M., and Leskovec, J. (2019). GNNExplainer: Generating explanations for graph neural networks. *Advances in Neural Information Processing Systems*, 32(iii).

Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., and Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI Open*, 1(January), 57–81.