Graph-Based Evolutionary Search for Optimal Hybrid Modularization of Building Construction Projects

Cao, Jianpeng; Said, Hisham; Savov, Anton; Hall, Daniel

DOI
10.1061/JCEMD4.COENG-14687

Publication date
2024

Document Version
Final published version

Published in
Journal of Construction Engineering and Management

Citation (APA)
Cao, J., Said, H., Savov, A., & Hall, D. (2024). Graph-Based Evolutionary Search for Optimal Hybrid Modularization of Building Construction Projects. *Journal of Construction Engineering and Management*, *150*(8), Article 04024098. https://doi.org/10.1061/JCEMD4.COENG-14687

Graph-Based Evolutionary Search for Optimal Hybrid Modularization of Building Construction Projects

Jianpeng Cao[1]; Hisham Said, A.M.ASCE[2]; Anton Savov[3]; and Daniel Hall, A.M.ASCE[4]

Abstract: Off-site construction has been a crucial part of industrializing the industry to realize higher productivity, better quality, and a more sustainable approach for constructing buildings. Off-site construction requires decomposing a floor plan into modules that can be in the form of either panelized walls or volumetric modules. However, the previous modularization models and approaches are limited due to their inability to consider the topological constraints of the modules, the flexible modularization of varying floor plans, and the mixed use of panelized walls and volumetric modules. As such, this paper proposes a graph-based optimization methodology for the hybrid modularization of building floor plans. The methodology was implemented using a multiobjective genetic algorithm that encodes and decodes the floor plan using novel graph modeling and operations. A visual programming script was developed to extract the wall properties, their adjacencies, and junction information from the building information model (BIM) of the floor plan. Time and cost estimation functions were developed to evaluate the hybrid strategies of panelized-volumetric modularization. The deployment of the methodology was demonstrated using an example floor plan design, which resulted in a spectrum of hybrid modularization plans ranging between fully volumetric and fully panelized solutions. For this specific example, the fully volumetric solution was 23% faster than the fully panelized solution but was 22% more expensive. The main contributions of this study are the topological modeling of module types, their floor plan postdesign flexible utilization, and the ability to explore hybrid modularization strategies. The findings of this study can prove useful for modular and off-site building manufacturers to improve their agility and increase their market share. **DOI: 10.1061/JCEMD4.COENG-14687.** © *2024 American Society of Civil Engineers.*

Author keywords: Modularization; Graph modeling; Volumetric module; Genetic algorithm.

Introduction

Modularization is one of the major construction industrialization tools to improve project overall productivity, enhance the delivered building quality, and reduce the environmental footprint of a business (Attouri et al. 2022). As a result, modularization adoption has increased in the industry and is expected to expand at a growth rate of 5.7% annually (Abdul Nabi and El-adaway 2020). In the design process, modularization refers to a building design to be decomposed into standardized modules [volumetric modules (VMs) or panelized walls (PWs)] (Feist et al. 2022), in order to handle the fabrication and logistical complexities of the supply chain.

The terms module and modularity have not been uniquely defined in construction literature (Gosling et al. 2016). Ulrich and Eppinger's (2018) definition is widely accepted in modular product design (Baldwin and Clark 2018; Duray et al. 2000; Jiao et al. 2007; Salvador et al. 2002; Stone et al. 2000). He described a module from two product properties: the similarity of the physical and functional structure, and decoupled interfaces between components (Ulrich and Eppinger 2018). The first property highlights the commonality in module design. A module is a group of product variants that share common features (Feist et al. 2022). The second property emphasizes the interdependence within and independence across modules. In that sense, a module is a unit whose elements are strongly connected among the elements in the same unit and relatively weakly connected to elements in other units (Samarasinghe et al. 2019).

The product modularization problem has been solved from two perspectives previously in the building construction industry. Firstly, modules are identified in a complex product. The solution defines the structural composition of modules and the number of modules (Isaac et al. 2016; Samarasinghe et al. 2019). Secondly, modules are modified by considering constraints in terms of design, transportation, manufacturing, and assembly. The geometric dimension of each module is then optimized (Almashaqbeh and El-Rayes 2021; Tidhar et al. 2021). The first scenario often takes place during conceptual design, whereas the second one is analyzed at the detailed design stage.

In industry practice, the modularization problem is usually solved based on the engineer's experience (Tidhar et al. 2021; Tserng et al. 2011). Engineers need to manually modularize a customized architectural design, often collaborating with architects and manufacturers. The approach is slow and usually leads to suboptimal solutions. Alternatively, designers select a set of standardized modules, usually developed by manufacturers, and apply it to their design. Although the construction process is sped up because the modules are well-engineered and known to be manufacturable, the design flexibility is limited, ending up with projects that

[1]Ph.D. Researcher, Institute of Construction and Infrastructure Management, ETH Zurich, Stefano-Franscini-Platz 5, Zurich 8093, Switzerland (corresponding author). Email: cao@ibi.baug.ethz.ch

[2]Associate Professor, Dept. of Civil Engineering, Santa Clara Univ., 500 El Camino Real, Santa Clara, CA 95053. ORCID: https://orcid.org/0000-0002-6601-9498. Email: hsaid@scu.edu

[3]Postdoctoral Researcher, Institute of Technology in Architecture, ETH Zurich, Stefano-Franscini-Platz 5, Zurich 8093, Switzerland. Email: savov@arch.ethz.ch

[4]Assistant Professor, Faculty of Architecture and the Built Environment, Delft Univ. of Technology, Julianalaan 134, Delft 2628 BL, Netherlands. Email: D.M.Hall@tudelft.nl

Note. This manuscript was submitted on October 27, 2023; approved on March 18, 2024; published online on June 12, 2024. Discussion period open until November 12, 2024; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Construction Engineering and Management***, © ASCE, ISSN 0733-9364.

sometimes suffer from a negative perception of their architectural qualities (Cao et al. 2021).

Therefore, there is a need for the development of computational approaches that can support the construction project team to modularize building in order to achieve operational efficiencies while satisfying the project's need for design freedom and flexibility. These two operational and design objectives are contradictory, and achieving an acceptable balance between them requires novel computational approaches that can consider hybrid approaches of modularization. A hybrid approach is needed to combine VM for the highly serviced and higher value parts, and to use long-span PW for the more open-plan areas (Lawson and Ogden 2008).

In this paper, we propose the development of a graph-based hybrid modularization optimization methodology for building floor plan designs. Graph modeling is used as an efficient approach to represent the floor layout data with the flexibility to add additional data of manufacturing and assembly (Isaac et al. 2016; Cao et al. 2022). Besides, modules, including VM and PW, are modeled as metagraphs. Building upon that, an evolutionary algorithm supported with graph operations is used to optimize the modularization of the floor plan into a set of PW and VM. The implementation and performance of the proposed methodology are illustrated using an application example.

Literature Review

In this study, a comprehensive literature review was conducted to determine the current state of knowledge regarding the optimization of modularization components and plans within the construction industry. Utilizing Google Scholar and Scopus as the primary search engine, keywords such as "modular design/construction," "module configuration," "modularization," "optimum modules," and "optimization" were systematically employed to identify relevant academic works. This search was refined to exclude studies outside the building sector. Besides, this research specifically focuses on studies presenting quantitative and computational models of modularization. In general, three computational strategies are suggested: independence-driven modularization, commonality-driven modularization, and index-driven modularization, as detailed in Table 1.

Independence-Driven Modularization

Independence-driven modularization aims to identify modules by maximizing the interaction in modules and minimizing the interaction between modules. For that purpose, a set of tools is developed, including the design structure matrix (DSM), graph-based approaches, the modular identification matrix (MIM), and the generational variance index (GVI). Among them, DSM has been widely used in the industry (Ahmadi et al. 2001; Sinha et al. 2020; Van Beek et al. 2010; Yu et al. 2007). Wee et al. (2017) examined the application of three modularization tools, namely DSM, MIM, and GVI, in a modular plant room case study. They found DSM to be the closest to the engineers' modularization mindset (Wee et al. 2017).

DSM is a network modeling tool used to model elements comprising a system and their interactions. A DSM is represented as a $N \times N$ matrix, representing the interactions among N elements (Eppinger and Browning 2018). Then, clustering algorithms are utilized to group elements into clusters by reordering the rows and columns of the DSM. Alternatively, graph-based approaches use nodes and edges to represent elements and interactions between elements respectively. Clustering algorithms are used to find subgraphs as modules.

Table 1. Quantitative and computational models of modularization

References	Systems	Modularization drivers	Computational algorithms
Aiello et al. (2012)	Floor plan	Material handling costs, aspect ratio, closeness, distance requests	Genetic algorithm
Schmidt et al. (2014)	Housing unit	Independence, Adaptability	Dependency structure matrix, clustering algorithm
Isaac et al. (2016)	Housing unit	Independence, replacement commonality	Graph-based model, clustering algorithm
Sharafi et al. (2017)	Modular building	Plan irregularity, energy efficiency, construction cost	Matrix-based method, integer programming
Said et al. (2017)	Panelized building	Panel commonality, Fabrication cost	Genetic algorithm
Wee et al. (2017)	Plant room	Independence	Dependency structure matrix, modular identification matrix generational variance index
Samarasinghe et al. (2019)	MEP systems	Independence, assembly cost, handling time	Dependency structure matrix, clustering algorithm
Almashaqbeh and El-Rayes (2021)	Floor plan	Construction cost, floor plan functional performance	Linear programming
Tidhar et al. (2021)	Floor plan	Cost, speed, quality	Greedy algorithm
Almashaqbeh and El-Rayes (2022)	Floor plan	Construction cost, floor plan functional performance transportation cost	Genetic algorithm
Feist et al. (2022)	Room modules	Room commonality	Clustering algorithm
Cao et al. (2022)	Framed panels	Panel commonality	Graph-edit distance algorithm
Ghannad and Lee (2023)	Floor plan	Module commonality, suitability	Genetic algorithm
Suárez et al. (2023)	Plumbing systems	Installation cost, local job creation	Dependency structure matrix, fuzzy logic models, genetic algorithm

Samarasinghe et al. (2019) and Suárez et al. (2023) adopted the DSM approach to modularizing mechanical, electrical, and plumbing (MEP) systems The approach does not take structural elements into account. Compared with modules in MEP systems, the structural modules are more restricted by design and installation constraints. Schmidt et al. (2014) adopted a DSM approach to analyze the interactions among the subsystems (e.g., skin) of a house. Within each subsystem, a combination of automated and manual clustering was undertaken to group building elements into modules. The elements in the detected modules might not be physically tied but functionally rely on each other (Schmidt et al. 2014). Isaac et al. (2016) carried out a graph-based approach to cluster MEP and structural elements into modules, and implemented the methodology in a case study of a robotic patient headwall system. They identified the modules with similar replacement rates during the operation stage of the building to minimize the renovation efforts.

In both of those studies, the identified module configuration may not be practical during the construction stage. The limitation is that no topological constraints were taken into account during module identification. Here, the topological constraints refer to the requirement of module composition and spatial relations between components. For example, the topological constraints for walls of a VM could be four-sided enclosed, open-ended on one side, or open-ended on opposite sides. Without taking topology as a constraint, the modules identified by the independence-driven approach might not be feasible in terms of structural integrity, transportation, and assembly.

Commonality-Driven Modularization

The commonality-driven modularization aims to optimize modules by detecting shared features among product variants and developing standard representatives. It is a postanalysis for typical VM and PW systems. The analysis starts with the product feature extraction. The next step is to calculate the difference between product variants using the extracted features. Finally, a clustering algorithm is implemented to group a number of product variants into clusters in which products are more similar to each other than in other clusters.

Previous studies applied this strategy to VM systems (Feist et al. 2022; Ghannad and Lee 2023) and PW systems (Cao et al. 2022; Said et al. 2017). The selected features can be summarized into three categories: architectural shapes and dimensions (Cao et al. 2022; Feist et al. 2022; Ghannad and Lee 2023; Said et al. 2017), element typologies (Cao et al. 2022; Feist et al. 2022), and structural composition (Cao et al. 2022; Feist et al. 2022). The clustering process is semiautomated, which requires manual selection of the number of clusters. By controlling the number of clusters, architects can make the trade-off between commonality and customization. Of the aforementioned studies, Said et al. (2017) proposed an optimization strategy to support architects to balance the trade-off by analyzing the commonality and fabrication cost simultaneously in panelized projects. Ghannad and Lee (2023) developed a systematic approach to balance the commonality and suitability of the module configuration in postdisaster housing projects. However, the commonality-driven approach lacks the ability to identify modules from different topologies, such as hybrid planar and volumetric buildings. Also, this approach does not support the entire modularization process, and rather focuses on the commonality and variety optimization of an already modularized system.

Index-Driven Modularization

The index-driven modularization is originally used as an evaluation approach to quantify the suitability of a module design solution in building construction. Boothroyd (1994) suggested that simplifying the structure with a minimum number of modules can reduce the total cost. In this regard, some studies considered the number of modules in determining the module configuration (Almashaqbeh and El-Rayes 2021; Liew et al. 2019). Smith (2010) discussed key constraints for module configuration, including transportation, assembly, crane, and tolerance limitations. Building upon that, Salama et al. (2017) introduced five quantifiable indices, including a construction index, transportation dimensions index, transportation shipping distance index, crane cost index, and concrete volume index, to enable a near-optimum selection of module configuration.

With the emergence of automated decision-making tools, a number of studies used computational algorithms to automate and optimize the modularization task. Aiello et al. (2012) optimized the design of facility layouts using multiobjective genetic algorithm optimization. However, they formulated the problem as a layout partition using rectangles without taking architectural requirements into account. To improve that, Sharafi et al. (2017) took plan irregularity, energy efficiency, and construction cost as objectives and generated floor plans using different shapes of units, such as T-shape, L-shape, linear shape, and rectangular shape. The modularization program was modeled as a three-dimensional assignment problem, and solved by integer programming.

Almashaqbeh and El-Rayes (2021) used linear programming to generate the optimal VM configuration for architectural floor plans by minimizing the total cost of modular construction projects and maximizing the functional performance of the floor plan. The optimal width and length of each room were identified as output. They further integrated transportation costs as another objective to determine the truck assignment plans. Tidhar et al. (2021) used a greedy algorithm to adjust the internal VM boundaries of a given architectural floor plan. They evaluate each design solution in terms of cost, speed, and quality. Both studies tackled the modularization problem by optimizing the dimensions of VM. However, the design was only limited to the application of a four-sided enclosed VM. There does not yet exist an approach for hybrid module solutions with diverse module topologies to be derived for customized architectural design.

The literature review revealed three main research gaps that need to be addressed by new modularization computational approaches. First, the topological constraints of the modules need to be considered during the modularization process. There is a need for modularization computational approaches that consider the functional and structural requirements of the modules, not just their clustered interdependencies (as done in DSM and independence-driven models).

Second, modularity should not be integrated too early in the design process in the form of standardized rigid modules, or too late in the process in the form of marginal commonality-focused design adjustments. There is a need for new models that allow the use of flexible modules that can be topologically defined and flexibly applied to a variety of floor plan designs.

Third, design modularization approaches have typically been limited to a single-product topology and failed to explore the benefits of integrating VM and PW to fit the needs of each individual project. There is a need for new approaches to allow the integrated use of panelized and volumetric modules to improve the design modularization process while considering the time and cost implications.

Research Goal and Methodology

To address the existing gaps in the literature, this paper presents the development of graph-based hybrid modularization optimization methodology of building floor plan designs while considering the topological constraints and the time-cost impacts of volumetric and panelized modules. Graphs are used frequently to represent design properties in creating building floor plans (Ślusarczyk 2018). During the last years, several graph-based methods have been proposed for the early-stage design optimizations and conceptual design decisions (Nauata et al. 2020; Pizarro et al. 2022; Sharma and Chattopadhyay 2018; Wang et al. 2018). Compared with traditional building information modeling (BIM), graphs allow for a more lightweight representation of buildings, focusing on the spatial configurations and supporting the design process through the integration of topology and semantics.

The proposed methodology adopts a graph-based approach for modeling the topological constraints of floor plan modularization. By doing so, the method enables integration of modularity during the design development stage, when the floor plan is given by designers, so as to avoid designs originating from too-rigid modules. Besides, instead of a single-product topology approach, diverse topologies of VM and PW products could be applied to modularize the building. The method benefits the stakeholders by leveraging a mass production approach in customized design, thereby achieving production-efficient solutions without sacrificing design flexibility.

Fig. 1 depicts the methodology, which involves the graph modeling of the project's floor plan and modular units (VM and PW), the genetic algorithm optimization of the project modularization strategy, and the visualization of the possible strategies and their modularized floor plans. The following sections thoroughly explain the proposed methodology.

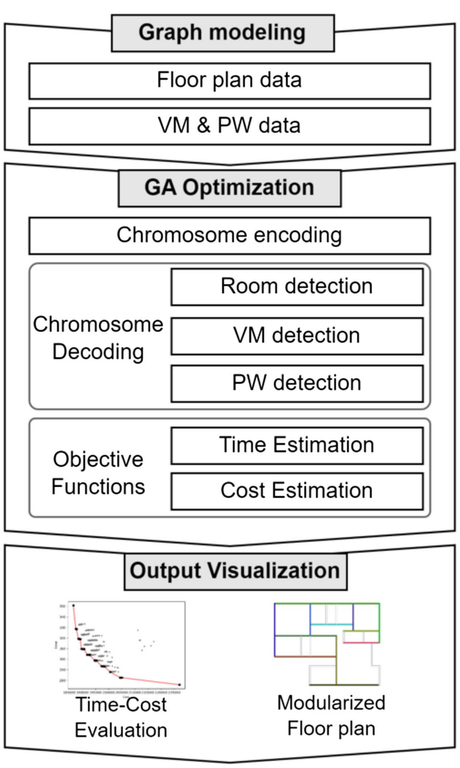

Fig. 1. Methodology of graph-based modularization optimization.

Graph Modeling

Floor Plan Representation

The floor plan was designed as a BIM, which contains architectural elements, structural elements in three-dimensional (3D) geometry. In this study, we applied an attributed graph to represent each floor plan. An attributed graph over the node set (N) and the edge set (E) is a system $G = (N, E, A_N, A_E)$ where N is the node set, representing a set of building elements, E is the edge set, representing a set of connection between elements, A_N is the attributes of nodes, representing the properties of the building elements, and A_E is the attributes of edges, representing the properties of the connection.

To build such a graph representation, the virtual programming tool Dynamo version 2.18.1 was used to extract information from the BIM. Two major operations were performed using Dynamo for graph development: node identity extraction and edge identity extraction. Dynamo is a visual programming scripting language. Unlike traditional programming, Dynamo uses a visual scripting interface with nodes and wires, making it accessible to those without a background in coding. Each node represents a function or action, and users can connect these nodes to construct complex logic and design algorithms.

Node Entity Extraction

This step is to extract the node identities for graph representation. Using Dynamo, elements were filtered with the Element Classes function. Element IDs are retrieved using Element.ID, and additional properties, such as length, are accessed through ParameterByName. This study focused on walls in a given floor. Wall segments of any length between wall intersections were modeled. These segments can be combined into a single PW, or left to be assembled onsite. Fig. 2 illustrates the Dynamo scripts utilized for extracting walls as nodes, including their IDs and lengths as node attributes.

Edge Entity Extraction

The edge identity was extracted from the BIM model in this process. Through Dynamo, the Element.Geometry and Geometry.DoesIntersect functions can be used to obtain the geometry models of building elements and determine connections between two geometry entities. Connections can be stored in a set of node pairs, <node_i, node_j>. In addition, the connection types and connection angles were extracted as edge attributes. In this study, we categorized two connection types, off-site and onsite connections, which will be explained in the section "Chromosome Decoding." The methods for calculating the connection angles between intersecting elements involve the Element.GetLocation and AngleWithVector functions, as depicted in Fig. 3.

Finally, the node and edge information can be transformed into a graph structure via NetworkX version 3.2.1 (Hagberg et al. 2008). NetworkX is a Python package for the creation, manipulation, and visualization of complex graph structures. A floor plan is transformed into a graph as an example (Fig. 4). Here, walls and connections between walls were modeled as nodes and edges, respectively. The wall types and dimensions were modeled as node attributes and connection angles as edge attributes.

Hybrid Module Representation

In the considered hybrid modular system, modules, including VM and PW, were modeled as metagraphs using the same modeling strategy as that for the floor plan. The VM units varied from complete rooms, parts of rooms, and separate highly serviced units such

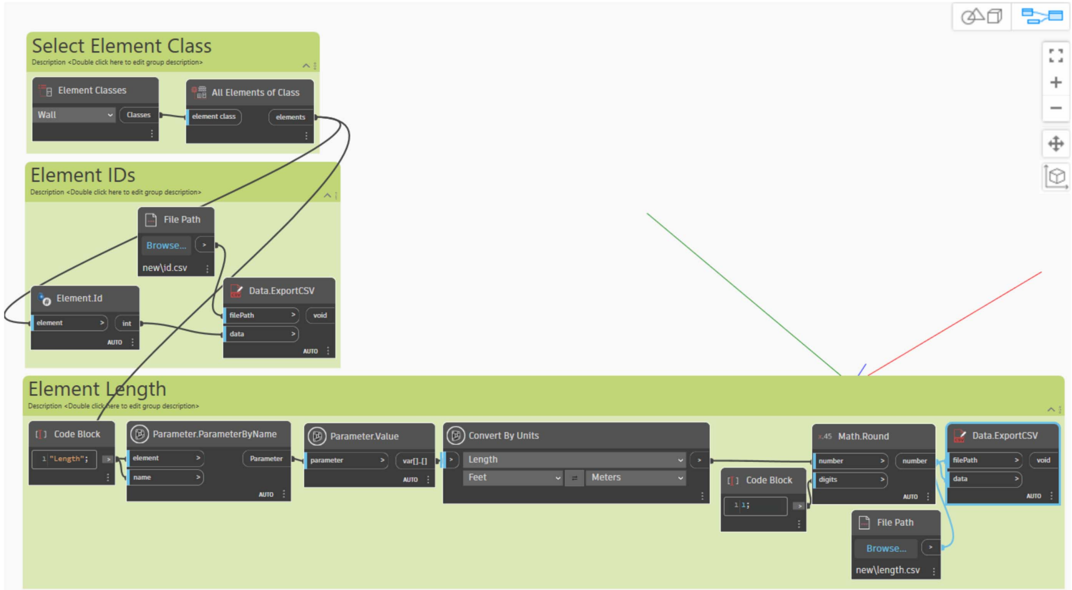

Fig. 2. Dynamo scripts for node entity extraction.

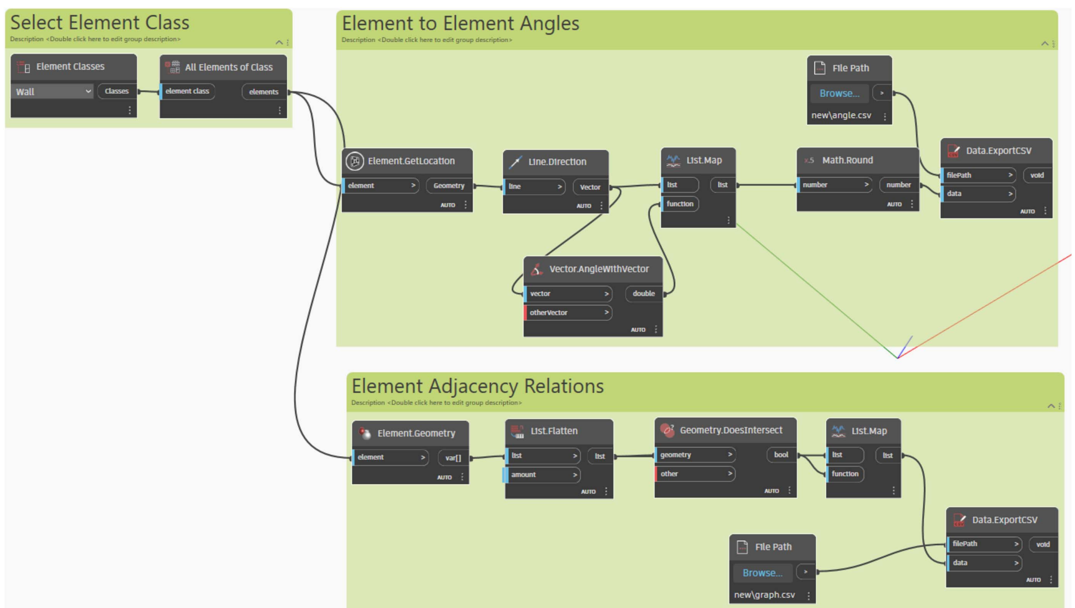

Fig. 3. Dynamo scripts for edge entity extraction.

as toilets or lifts. The load-bearing wall panels and floor cassettes create a more flexible open space (Lawson et al. 2014).

Volumetric Modules

In this study, two topology types of VM were considered in the design of hybrid systems: four-sided VM and partially open-sided VMs. The four-sided VM are manufactured off-site from a series of two-dimensional (2D) wall segments, starting from the floor cassette, to which the four walls and ceiling are attached. The four-sided VM is suitable for various structural material types, such as concrete, steel, and timber structures. They are prefabricated off-site, transported to the site, and craned onto the building foundation or on top of an assembled floor. The four-sided VM can be manufactured without a floor and ceiling structure, such as for lift shafts. Taking a four-sided VM in Fig. 5 as an example, six building elements, namely, four walls, a ceiling, and a floor, were modeled as nodes. Connection angles (any integer value between 0° and 180°) and connection types (onsite or off-site) were modeled as edge attributes.

Partially open-sided VM improve the design flexibility because two VM can be placed together to create wider spaces. In this case, one of the walls in the longer orientation of the module is removed or partially included in the module to create an open area on one side. The other longitudinal wall is needed to handle the module lateral loads, and the vertical loads are supported via the other two transverse walls. Combining two partially open-sided VM units could satisfy the typical room sizes of residential floor plans. The graph representation is shown in Fig. 6.

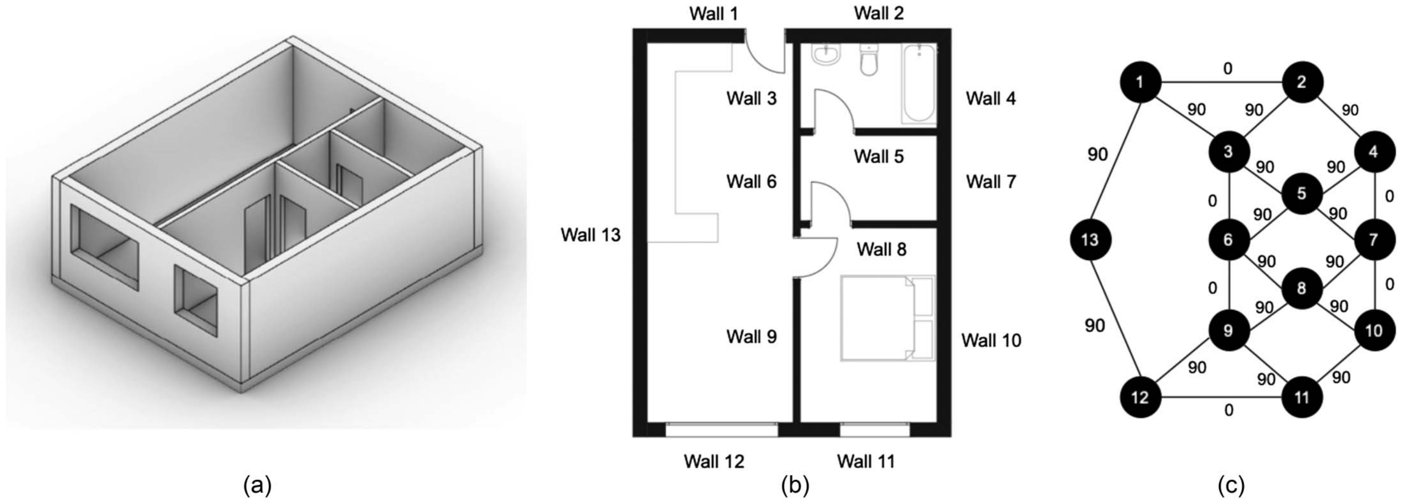

Fig. 4. Floor plan and corresponding graph representation: (a) 3D model of the floor plan; (b) 2D view of the floor plan with identifiers of wall segments; and (c) graph representation where the nodes are mapped to the wall segments in plot (b), and the edge value, either 0 or 90, indicates whether a wall is parallel or perpendicular to another wall.

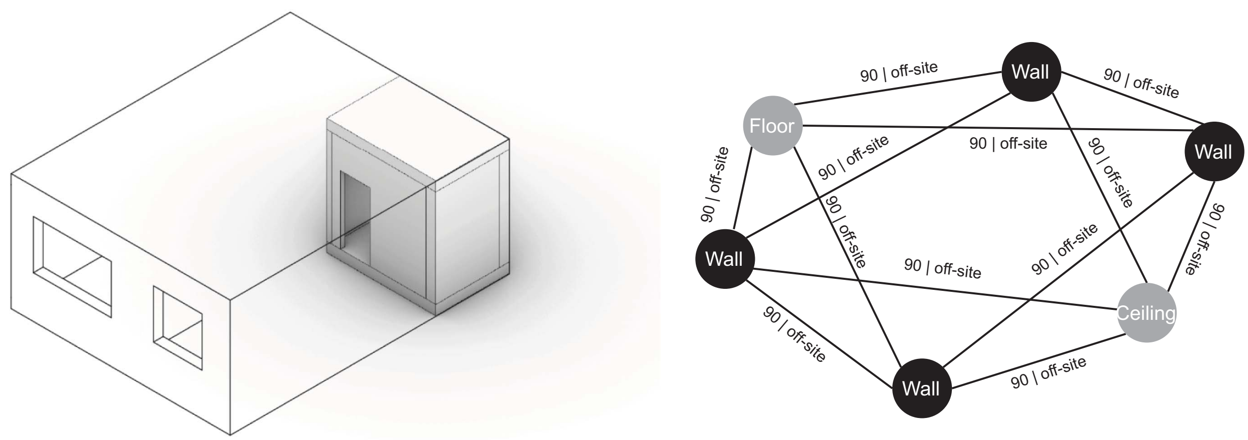

Fig. 5. Four-sided VM and corresponding graph representation.

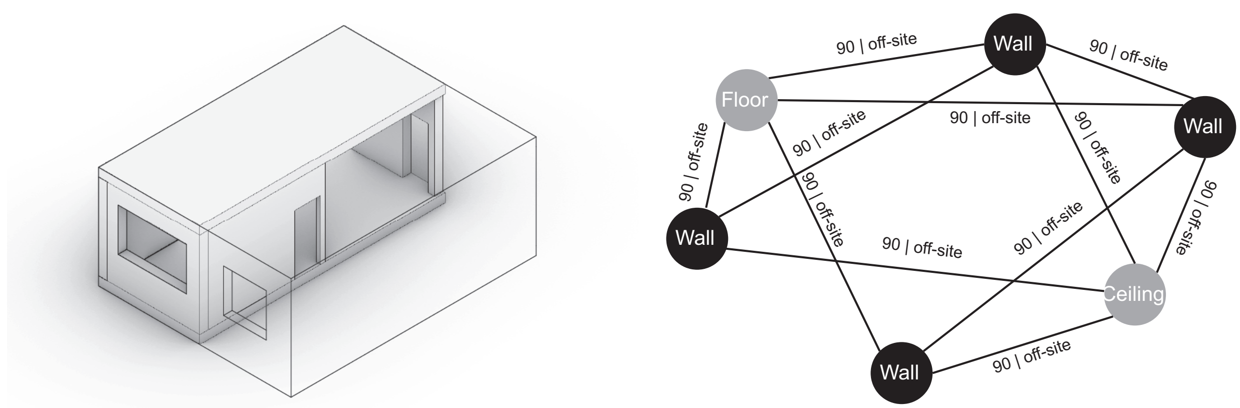

Fig. 6. Partially open-sided VM and corresponding graph representation.

PW Panels

Two-dimensional flat panels can be used as exterior facades and interior partitions, which can be prefinished with insulation and boarding before delivery to the site. Initially, walls are segregated into wall segments between their intersections. Each wall segment is represented by a single node in the floor plan graph. The proposed approach considered merging these wall segments into continuous wall panels. Panel length can be maximized so as to reduce the number of panels to be manufactured and lifted, with the aim to improve overall productivity by reducing the non-value-adding time of fabrication setup and handling. The maximum panel length is usually controlled by special transportation permits. Two segments can be merged if the connection angle between them equals 0° or 180°, and two walls share the same material types. The conditions for merging can be easily extended.

Genetic Algorithm for Modularization Problem

To account for the trade-off between minimizing both the construction cost and time, the floor plan modularization analysis was formulated as an optimization problem of value assignment for edge properties (i.e., connection types) of the graph model of the floor plan. This optimization problem was implemented using a multiobjective genetic algorithm (Golberg 1989) that encodes the edge list of a graph as a chromosome and takes the total floor construction time and cost as the fitness function. The optimal solution can be decoded to a modularized floor plan that consumes the least time and cost. Genetic algorithms (GAs) are widely used to generate near-optimal solutions in the architectural and structural design domains (Anvari et al. 2016; Bianconi et al. 2019; Yazdi et al. 2021; Wang et al. 2016). GA is one of the common evolutionary algorithms that is simple to implement to optimize nonlinear problems, but it can be computationally demanding and requires fine-tuning of its parameters (Val et al. 2021). In addition, GA is best suited for multiobjective optimization problems where the decision maker needs to analyze the possible trade-offs among the considered objectives (Said and El-Rayes 2013).

To use GA as a solution search method, the design variables must be encoded as chromosomes, where each chromosome includes multiple genes that represent the optimization decision variables. A chromosome is composed of a list of the connection types between the wall segments as design binary variables. Each variable has two categorical values: 1 represents and off-site connection and 2 represents an onsite connection. The length of the chromosome is the number of connections in the given floor plan, which equals the number of edges of the floor plan graph representation. Fig. 7 depicts a small floor plan example of two rooms, seven wall segments (graph nodes), and 10 wall connections (graph edges). As such, this floor plan modularization problem will be encoded as a 10-gene chromosome, where each gene represents a wall connection type. Using these connection type decision variables, the floor plan of discretized wall segments can be grouped into a hybrid set of VM units and PW two-dimensional panels during the decoding process.

As shown in Fig. 7, the algorithm started by creating a new population of chromosomes by randomly setting the chromosome gene value to sample the design search space. As such, this population and subsequent populations were evolved and optimized iteratively in the following steps (Deb et al. 2002): (1) decoding the chromosome using graph operations of room search, VM search, and PW search (explained subsequently), (2) evaluating the fitness of each population chromosome using the time and cost objective functions (explained subsequently), (3) sorting the fitness of each chromosome by its dominances of other chromosomes in terms of lower time and cost values, (4) generating a new population by applying the GA evolutionary operations of parent selection, crossover, mutation, and elitism [Deb et al. (2002) has given a full description], and (5) checking if the solution convergence termination condition is satisfied to stop the algorithm and output the final population and its nondominated optimal chromosomes (modularization plans). The following sections describe in more detail the chromosome decoding and optimization objective functions of the modularization problem.

Chromosome Decoding

Three graph operations, including room detection, VM detection, and PW detection, were performed sequentially on the graph model of the analyzed floor plan to decode the chromosome values and to translate them into a modularization solution. Two constraints were applied in these operations to penalize any modularization solution that exceeded the maximum panel length and the maximum VM volume limits.

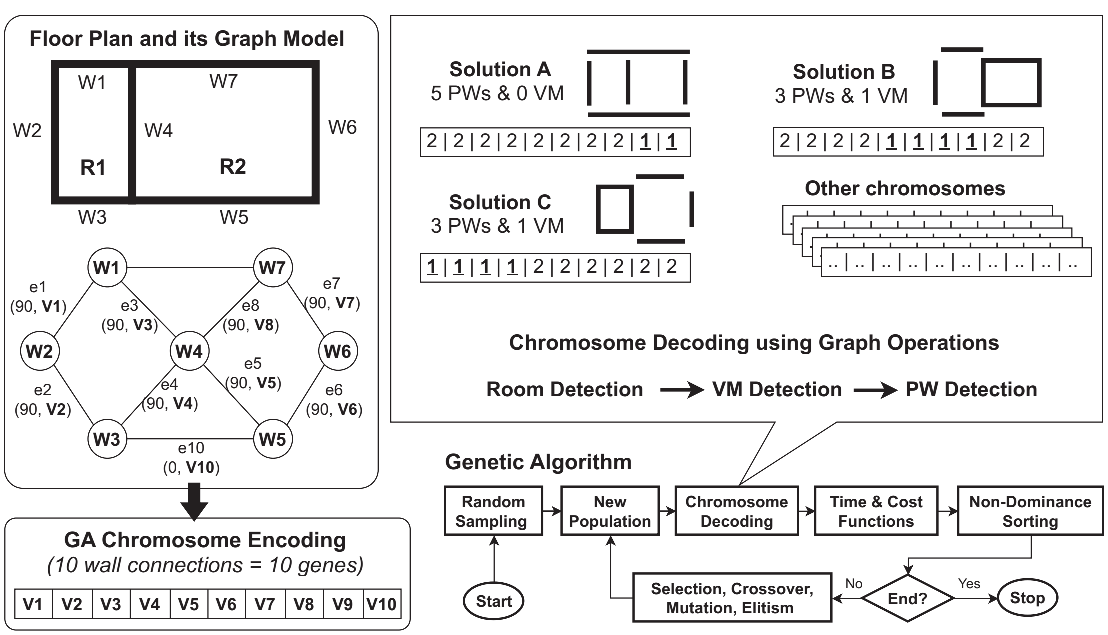

Fig. 7. Implemented genetic algorithm and its graph-based chromosome encoding and decoding.

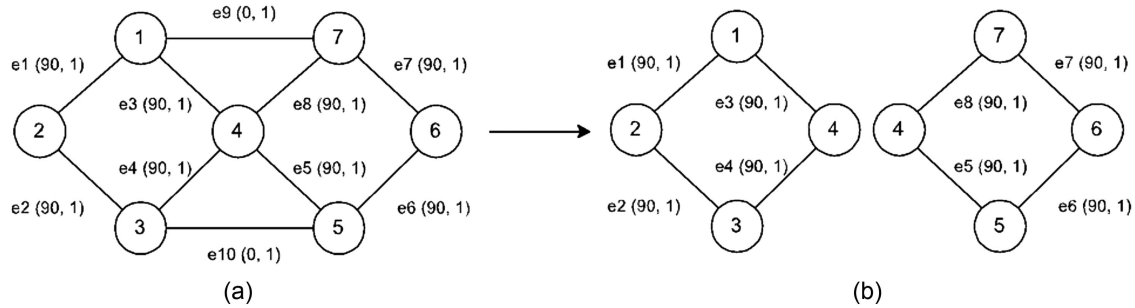

Fig. 8. (a) Nonsimple cycle; and (b) two simple cycles.

Room Detection

A room is an enclosed space unit in a given floor plan. Each room can be constructed differently according to its shape, scale, and relationship to other rooms. Enclosed rooms are detected in the floor plan by finding all simple cycles. A cycle is a simple cycle if it cannot be broken down into two or more cycles. In this sense, a simple cycle represents an enclosed room that is not partitioned into more rooms. To find all simple cycles in a graph, we used a minimum cycle basis (MCB) algorithm (Mehlhorn and Michail 2009). The algorithm is used to find cycles that the sum of the weights of edges is minimal. For unweighted edges, the sum of the weights of edges equals the number of edges.

Based on the algorithm's results, a filtering operation was conducted to identify the cycles with the number of edges (walls) to be more than three and the number of 90° connections to be exactly four. The first criterion aims to avoid detecting a cycle that represents an intersection between three wall segments (e.g., 1-4-7 and 3-4-5 in Fig. 8). The second criterion is used to find rooms in a rectangular shape, which is common for VM (e.g., 1-2-3-4 and 4-5-6-7 in Fig. 8).

VM Detection

For each detected room, the volume of the room was compared with the maximum VM volume limit to check whether the room can be modularized or not. Then, the VM topology was determined at this step via subgraph matching. Subgraph matching was done to determine the presence of a given query graph in a target graph (Bonnici et al. 2013) (Fig. 9). If the graph contains the node and edge properties, both the topology and the properties should be matched. In this study, we represented the user-defined VM as metagraphs and used them as query graphs to find VM in a target floor plan graph. The node properties, including element types and edge properties, including angles and connection types, were taken into matching process. The connection types, specifically, are dependent on the GA chromosome values.

In detail, a query subgraph is represented as a sequence of connected nodes and edges in a form of (typenode_1, angleedge_1, connection typeedge_1, typenode_2), (typenode_2, angleedge_2, connection typeedge_2, typenode_3), ..., (typenode_x, angleedge_n, connection typeedge_n, typenode_y). Because the query graph represents a VM, the connection type of all the edges in the matched subgraph has to be an off-site connection.

PW Detection

After finishing the VM detection, the unmatched part of floor plan was constructed using panels. The length of panel can be maximized by merging multiple wall segments into one within the predefined length limitation. The merging of wall segments was performed using a node contraction operation. The contraction of a pair of Nodes 1 and 2 of a graph is the operation that produces a graph in which Nodes 1 and 2 are replaced with a single node v, such that v is adjacent to the union of the nodes to which Nodes 1 and 2 were originally adjacent. This operation requires that the wall angle between merging segments is zero. Furthermore, node contraction iteratively continues until no further wall mergers are possible. An example of node contraction is shown in Fig. 10.

When two nodes are contracted as one, the edge property and node property are updated. For example, the Length property of the replaced single node is recalculated as the sum of the length property values of the original two nodes. The model checks the updated length property of the merged PW condensed node to confirm that it is within the maximum allowable wall lengths permitted by the shipping trailer dimensions.

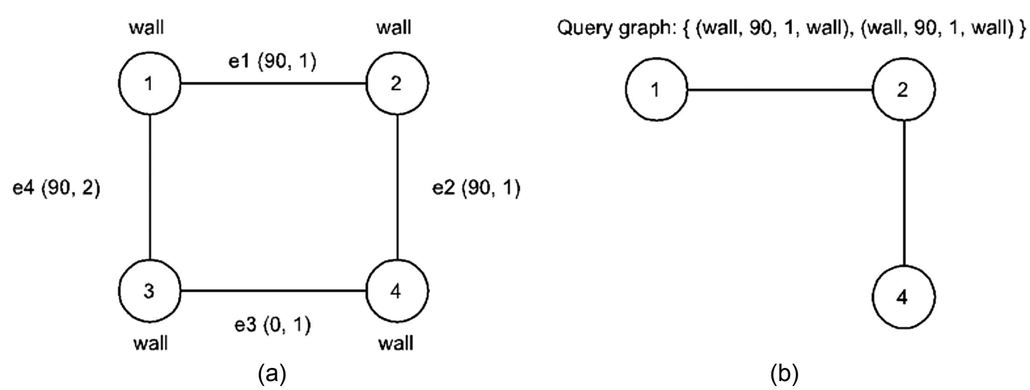

Fig. 9. (a) Target graph; and (b) detected query graph.

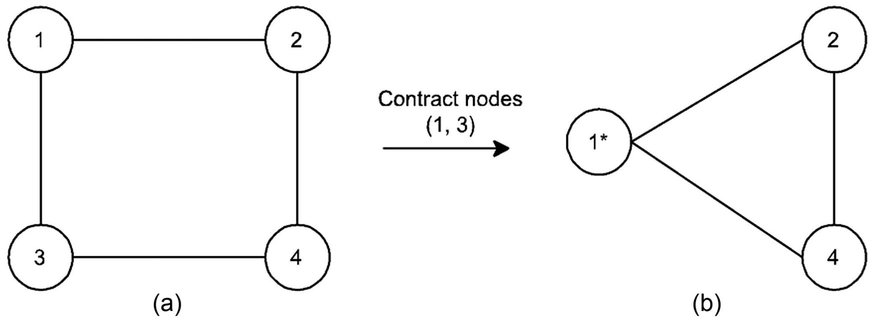

Fig. 10. Node contraction on (a) original graph; and (b) output graph.

Objective Functions

The evaluation of a modularized solution was based on the total floor construction duration (TD) and total cost (TC). The construction total duration calculation considers the main activities of off-site fabrication, onsite assembly, and interior finishes. Other activities were not considered due to their short time spans or independence from the selected modularization approach (like the facade). The total cost was calculated as the sum of the fabrication, finishing, assembly, and shipping costs. The goal of the developed model is to generate a modularization solution that simultaneously minimizes both objective functions, or generate Pareto solutions that provide different trade-offs between the objectives of minimizing cost and time. We define the notations, parameters, functions and calculation variables in the "Notation" section.

The proposed calculation functions attempt to provide approximate duration and cost estimates of the volumetric and panelized fabrication and installation processes. Similar estimate assumptions and approximation approaches were made in previous studies (Khalili and Chua 2014; Anvari et al. 2016; Yang et al. 2016), which did not limit the practicality and relevance of their developed models.

Time Estimation

The construction time of the hybrid modularized floor was estimated using a linear production modeling approach to account for the concurrent and continuous execution of the fabrication and assembly tasks. As shown in Fig. 11, the construction scope involved six main processes: panel fabrication for the PW scope, panel fabrication for the VM scope, module fabrication, panel assembly, module assembly, and site finishes.

Fig. 12 depicts the linear production model used to parametrically estimate the construction total duration based on the durations and the buffers of the construction processes, which are calculated based on the analyzed modularization solution. The mathematical representation of the construction time estimation is presented in Eqs. (1)–(11), which are formulated considering the following assumptions and operation logics:

- As shown in Eq. (1), the construction total duration was calculated as the sum of the total fabrication time (TFB) and total assembly and finish time (TAF). A decoupling point was placed between the fabrication and assembly to follow the logic that the assembly tasks of the VM units and PW panels commence at the same time.
- The fabrication of the VM units and the PW panels is done in the same factory, and their total fabrication time is modeled by Eq. (2). Thus, there are two panel fabrication processes: the first process (PFT_1 duration) fabricates panels for the VM scope and the second process (PFT_2 duration) fabricates panels for the PW scope. Both panel fabrication processes depend on the panel unit length fabrication time (PLT) and the length of each panel $L(P)$ used for PW units $N(P_{2D})$ and VM units $N(P_{3D})$. VM scope panels are fabricated before the PW scope panels to allow the immediate start of the module fabrication process, which involves the module interior finishes. A fully volumetric prefabrication strategy can benefit from this concurrent fabrication of PW and VM units (including the room finishes) to provide the greatest time savings.
- Three types of the connection were considered, including panel-to-panel connections, panel-to-VM connections, and VM-to-VM connections.
- The VM fabrication takes place off-site and its time (MFT) includes panel lifting, panel-to-panel connection, and VM finishes, as shown in Eq. (6). The whole off-site VM fabrication process is more efficient than onsite activities. Thus, an off-site relative unit rate factor (β) was used for MFT calculation, where a lower value β implies more efficient off-site operations that would require less time and cost per unit of work. As such, a fully volumetric approach can benefit from this off-site

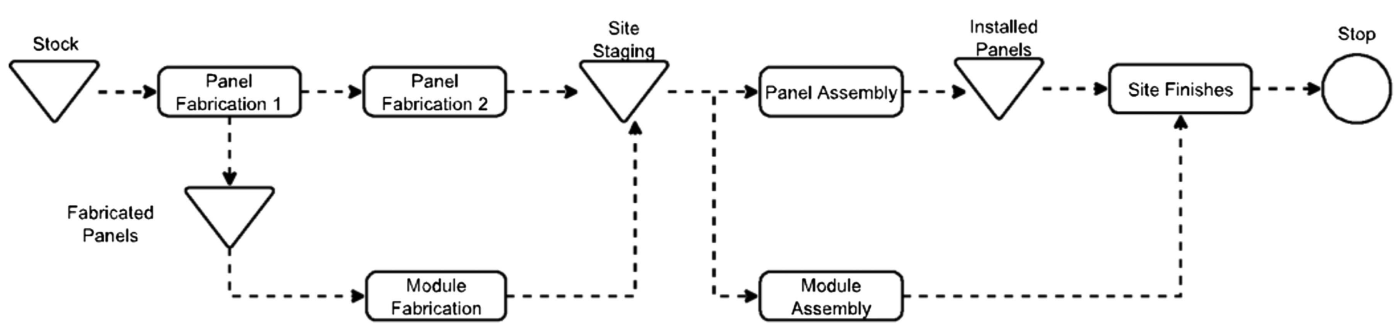

Fig. 11. Processes of the hybrid module construction.

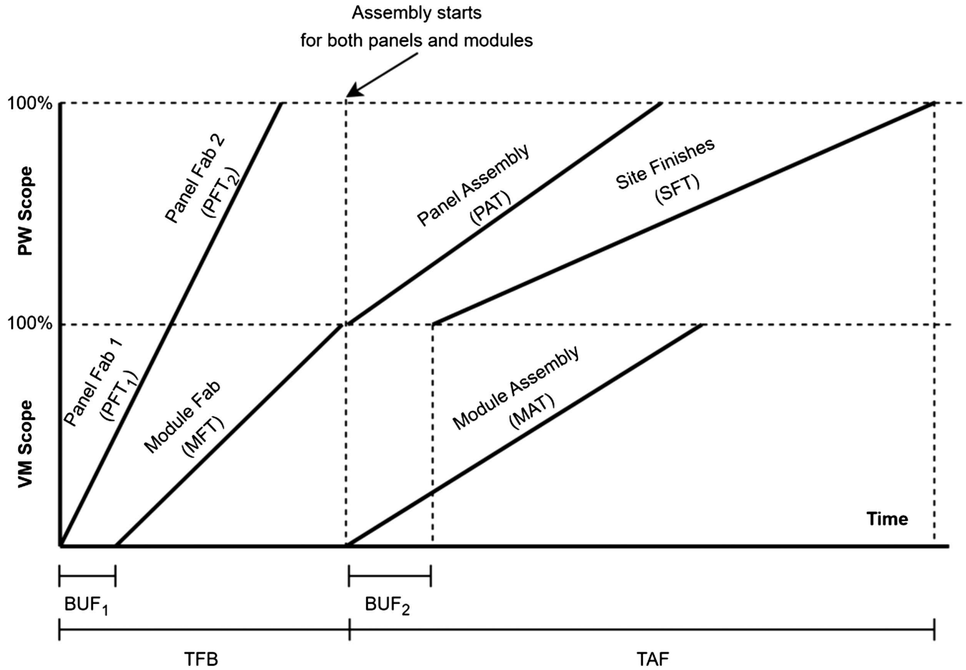

Fig. 12. Linear production representation of the fabrication and assembly of the floor VM units and PW panels.

production efficiency advantage by achieving greater time savings in both fabrication and finish work.

- The panel assembly takes place onsite and its time calculation (PAT) includes panel lifting (CRNT_p), panel-to-panel connection (CONT_{p2p}), and panel-to-VM connection (CONT_{p2m}), as shown in Eq. (7).
- The VM assembly takes place onsite and its time calculation (MAT) accounts for VM lifting (CRNT_m) and VM-to-VM connection (CONT_{m2m}), as shown in Eq. (8).
- As shown in Eq. (9), the time of the site finishes (SFT) involve finishing work for nonvolumetric rooms (nr), which represents mechanical, electrical, plumbing, painting, flooring, insulation, openings, and similar work other than the walls. The time for each room (FT_r) is defined by user inputs to reflect varying levels of finish work for dry and wet rooms.
- Buffer times were used to allow the concurrent process execution within the fabrication and site phases of the construction project. Eq. (10) calculates the buffer time (BUF_1) between the panel fabrication and the VM assembly processes, and Eq. (11) calculates the buffer time (BUF_2) between the panel assembly and the site finishes. Here, it was assumed that both buffer times BUF_1 and BUF_2 are linearly correlated with the corresponding production rate of their preceding activities (PFT_1 and PAT, respectively), where the linear coefficients (STK_1 and STK_2) could be set by fabrication companies to reflect the minimal required buffers of panels for fabricating or assembling a module

$$\mathrm{TD} = \mathrm{TFB} + \mathrm{TAF} \tag{1}$$

$$\mathrm{TFB} = \max((\mathrm{PFT}_1 + \mathrm{PFT}_2), (\mathrm{BUF}_1 + \mathrm{MFT})) \tag{2}$$

$$\mathrm{TAF} = \max((\mathrm{BUF}_2 + \mathrm{SFT}), \mathrm{MAT}) \tag{3}$$

$$\mathrm{PFT}_1 = \sum_{p=1}^{(P_{3\mathrm{D}})} L(p) \times \mathrm{PLT} \tag{4}$$

$$\mathrm{PFT}_2 = \sum_{p=1}^{N(P_{2\mathrm{D}})} L(p) \times \mathrm{PLT} \tag{5}$$

$$\mathrm{MFT} = \sum_{m=1}^{N(M)} (N(P_m) \times \mathrm{CRNT}_p + N(p2p) \times \mathrm{CONT}_{p2p} + \mathrm{FT}_m) \times \beta \tag{6}$$

$$\mathrm{PAT} = N(P_{2D}) \times \mathrm{CRNT}_p + N(p2p) \times \mathrm{CONT}_{p2p} + N(p2m) \times \mathrm{CONT}_{p2m} \tag{7}$$

$$\mathrm{MAT} = N(m) \times \mathrm{CRNT}_m + N(m2m) \times \mathrm{CONT}_{m2m} \tag{8}$$

$$\mathrm{SFT} = \sum_{r=1}^{N(nr)} \mathrm{FT}_r \tag{9}$$

$$\mathrm{BUF}_1 = \mathrm{STK}_1 \times \frac{\mathrm{PFT}_1}{N(P_{3\mathrm{D}})} \tag{10}$$

$$\mathrm{BUF}_2 = \mathrm{STK}_2 \times \frac{\mathrm{PAT}}{N(P_{2\mathrm{D}})} \tag{11}$$

Cost Estimation

A parametric approach was used to estimate the construction total cost using hourly cost rates and area cost rates of the fabrication, shipping, assembly, and finishing steps for the considered modularization solution. Eqs. (12)–(16) were formulated to calculate TC based on the following reasoning and assumptions:

- As shown in Eq. (12), TC is made up of fabrication cost, finishing cost, assembly cost, and shipping cost.
- The fabrication cost includes wall panel cost (\$/m) and floor panel cost (\$/m^2). Both the wall panels and floor panels include the panels used for VM and panels for nonvolumetric rooms. The fabrication cost calculation in Eq. (13) accounts for the doubled floor panels usage in a volumetric room by duplicating

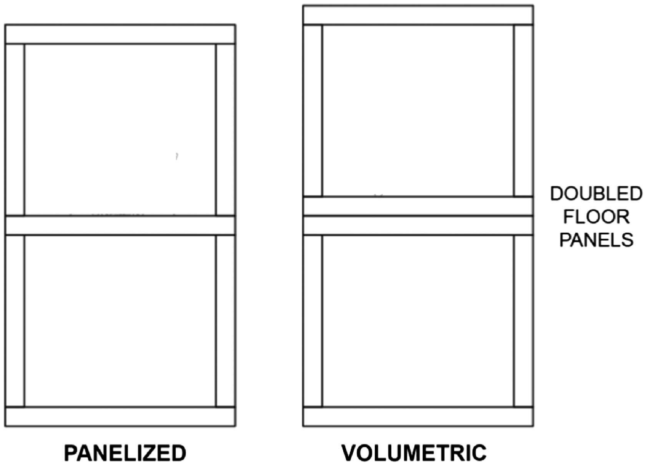

Fig. 13. Floor panel duplication in VM.

the area of such a room A(m), as shown in Fig. 13. A fully panelized prefabrication approach would provide material and fabrication savings by avoiding the need to duplicate the walls between volumetric modules.

- The finishing cost is calculated using Eq. (14) for the VM and nonvolumetric rooms. Considering the former work is done off-site, an off-site relative unit rate factor (β) is multiplied by the finishing cost rate (FC_m) for VM. As such, a fully volumetric approach results in the least finishing cost due to the off-site production efficiency advantage.
- The assembly cost is estimated using Eq. (15) by multiplying the assembly times of the panels (PAT) and the modules (MAT) with their assembly crew cost rates, PACR and MACR, respectively.
- The shipping cost is estimated by the number of trucks needed and trucking round-trip cost (TRC). A truck can take either a VM unit or its truck capacity load (TCP) of stacked panels. From a shipping-cost perspective, a fully panelized prefabrication strategy is favorable by saving on the truck trips

$$\mathrm{TC} = C_{\mathrm{fab}} + C_{\mathrm{finish}} + C_{\mathrm{assembly}} + C_{\mathrm{ship}} \tag{12}$$

$$C_{\mathrm{fab}} = \left(\sum_{m=1}^{N(m)} \sum_{p=1}^{N(P_m)} L(p) + \sum_{p=1}^{N(P_{2D})} L(p) \right) \times \mathrm{PLC} + \left(2 \times \sum_{m=1}^{N(m)} A(m) + \sum_{r=1}^{N(nr)} A(r) \right) \times \mathrm{FLFC} \tag{13}$$

$$C_{\mathrm{finish}} = \sum_{m=1}^{N(m)} A(m) \times \mathrm{FC}_m \times \beta + \sum_{r=1}^{N(nr)} A(r) \times \mathrm{FC}_r \tag{14}$$

$$C_{\mathrm{assembly}} = \mathrm{PAT} \times \mathrm{PACR} + \mathrm{MAT} \times \mathrm{MACR} \tag{15}$$

$$C_{\mathrm{ship}} = \left(N(m) + \frac{N(P_{2D})}{\mathrm{TCP}} \right) \times \mathrm{TRC} \tag{16}$$

Implementation and Performance Demonstration

The proposed methodology was implemented as an Autodesk Revit version 2024 plug-in and demonstrated via an illustrative case study of a residential project. The plug-in was developed in C# within Visual Studio 2019 from scratch. A conceptual unified modeling language (UML) class diagram for the plug-in design is shown in Fig. 14.

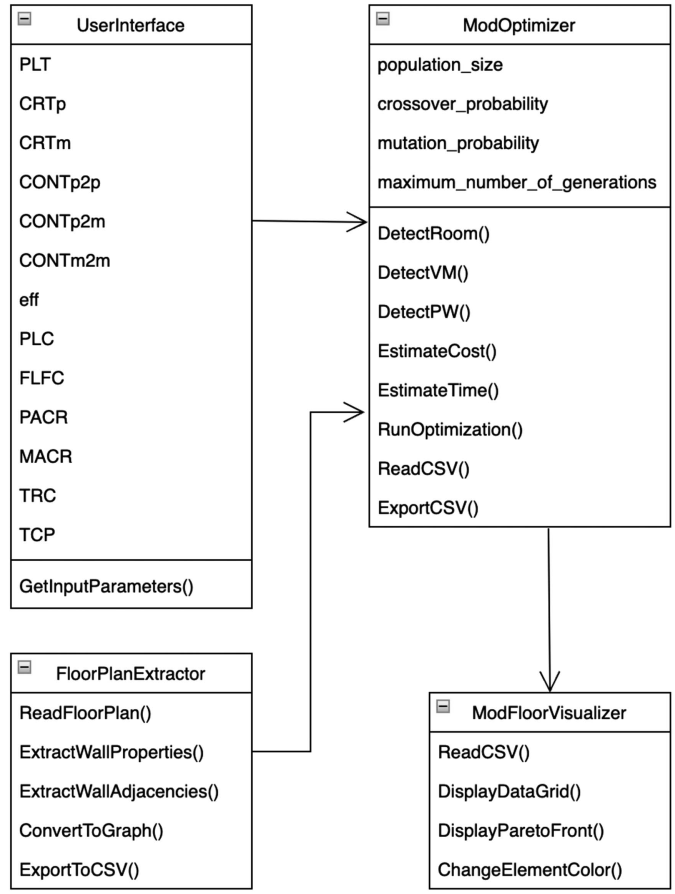

Fig. 14. UML class diagram of the plug-in design.

First, manufacturing process and cost parameters were entered as input. Second, the floor plan data, including wall properties and wall-to-wall adjacencies and angles, were extracted using Dynamo scripts and then converted into graph models using NetworkX, with nodes representing walls and edges denoting wall-to-wall connections. The attributes of nodes indicate wall properties, and the attributes of edges represent the angles between connections. Third, the plug-in fed user inputs regarding the manufacturing process, cost parameters, and floor plan data to the graph-based genetic algorithm to retrieve the optimal modularization solutions. The Revit application programming interface (API) was used to locate elements using Autodesk.Revit.DB.ElementId and color the VM units and PW units separately using Autodesk.Revit.DB.OverrideGraphicSettings. Users can view each generated solution in detail because its module assignment geometry can be visualized in real time, as shown in Fig. 15.

Due to the lack of similar hybrid modularization methodologies, a hypothetical project was introduced in this study to validate the proposed methodology by assessing the confidence in its causal modeling relations. Fig. 16 represents the floor plan of the case study, which resembles a residential unit design obtained from the Internet. The floor plan exhibits some complexity of real-world projects within a limited size, which aligns with the verification and face validation approaches (Lucko and Rojas 2012) in existing literature (Shewchuk and Guo 2012; Anvari et al. 2016; Almashaqbeh and El-Rayes 2021).

The project was modeled in Revit, where the rooms were classified as three wet rooms and nine dry rooms. The design information

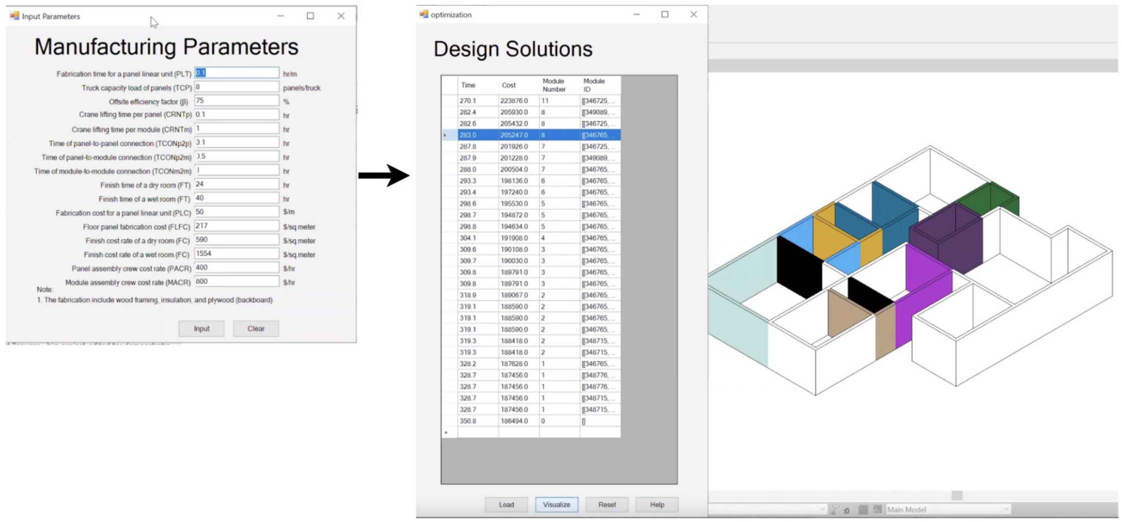

Fig. 15. Graphical user interface (GUI) of the plug-in.

Fig. 16. Residential floor plan of the application case study.

included room areas, room types, wall IDs, wall length, and wall-to-wall connection angles, which were extracted from Revit. The input values of the manufacturing process and cost input parameters are given in Table 2. Although these numbers are meant for illustrative purposes, they were average estimates of the observations from a set of industry references and various literature reviews. The cost-related parameter values were based on RSMeans detailed building data (RSMeans 2018) and unit area cost data (RSMeans 2022).

On the other hand, the time-related parameter values were based on a set of previous studies that analyzed the production of panelized and volumetric fabrication and installation processes (Shewchuk and Guo 2012; Ayinla et al. 2022; McGraw-Hill Construction 2011; Forsythe and Sepasgozar 2019; Boafo et al. 2016; Xu et al. 2020; Duncheva and Bradley 2019). Future modular builders could replace them with their own specific costs and productivity rates to generate results relevant to their own context.

The genetic algorithm was encoded in the backend using the Pymoo package version 0.6.1 (Blank and Deb 2020b). Following the suggestions given by Deb et al. (2002), the following values were set for the genetic algorithm run parameters: population size $= 400$, crossover probability $= 0.9$, mutation probability $= 1/40$, and maximum number of generations $= 100$. A running metric (Blank and Deb 2020a) was used to check the convergence of the genetic algorithm search by tracking the difference in the objective space from one generation to another. The detailed intermediate and final output, including the decoded solution chromosomes and their objective functions, were saved in a log file for verification and reporting purposes. The model verification was performed internally by the researchers (Lucko and Rojas 2012) by checking the log files and visualizing the results (modularized floor plans and the Pareto curve of the optimal solutions) to confirm the correctness of the model procedures and calculations.

Results

The solutions were plotted using a Pareto front graph (Fig. 17) to visualize the trade-offs between minimizing the construction duration and cost. From the Pareto front, the shortest-time solution was the fully volumetric solution [Fig. 18(a)], where 11 VM units were used to construct all rooms that match the topology of the considered parametric module. The lowest-cost solution utilized a fully panelized approach [Fig. 18(b)], where straight wall segments were merged as panels and no VM units were used.

Following the Pareto front from the top left corner to the bottom right corner, solutions with faster durations can be achieved by utilizing more VM units to construct the given floor plan in an optimal manner. The Pareto front curve followed a ladder shape, where each drop represents the addition of a VM unit to the modularization strategy. After a VM is added, other optimal modularization strategies were differentiated by (1) changing the topology type of the VM units (four-sided or partially open-sided), and (2) for partially

Table 2. Input parameters of the case study

Input parameter description	Input parameter	Value	Unit
Capacity of fabricating panel	PLT	0.1	Hours/meter
Truck capacity load of panels	TCP	8	Panels/truck
Off-site relative unit rate factor	β	75	Percentage
Crane lifting time per panel	CRNTp	0.1	Hours
Crane lifting time per module	CRNTm	1	Hours
Time of panel-to-panel connection	CONTp2p	0.1	Hours
Time of panel-to-module connection	CONTp2m	0.5	Hours
Time of module-to-module connection	CONTm2m	1	Hours
Finish time of a dry room	FT	24	Hours
Finish time of a wet room	FT	40	Hours
Fabrication cost for a panel linear unit	PLC	50	$/meter
Floor panel fabrication cost	FLFC	217	$/square meter
Finish cost rate of a dry room	FC	590	$/square meter
Finish cost rate of a wet room	FC	1,554	$/square meter
Panel assembly crew cost rate	PACR	400	$/hour
VM assembly crew cost rate	MACR	800	$/hour
Trucking round-trip cost	TRC	950	$/truck-trip
Panel stocks	STK_1 and STK_2	4	Panels

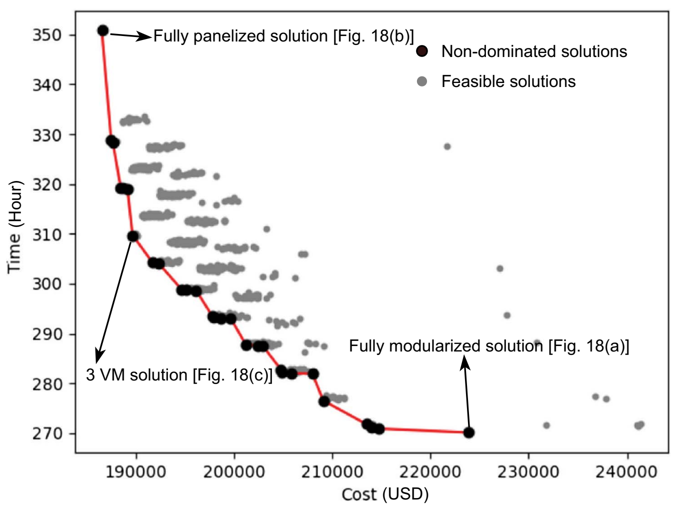

Fig. 17. Pareto front view.

open-sided VM units, changing the location of the module open side. The algorithm completes the modularization plan of each solution by merging the wall segments of the nonvolumetric rooms into PW units according to the shipping constraints. For example, when the number of VM equals three, the algorithm can identify which three rooms should be modularized first. In this case, the VM units were assigned to the three bathrooms in the floor plan [Fig. 18(c)]. The model prioritized the bathrooms to modularize as VM units due to their high finishing cost ($1,554/m^2) compared with other dry spaces.

For this specific illustration project case, the trade-offs between time and cost for modularized solutions are analyzed in more detail. Fig. 19 depicts the change of the contribution of the off-site work (fabrication) and onsite work (assembly and finishing) to the total duration over the modularization strategy spectrum. The contribution of the fabrication to the total duration dropped from 96% in the fully volumetric solution to 10.4% in the fully panelized solution. This is due to the fact that less work is externalized to off-site fabrication when the project scope is modularized into less volumetric

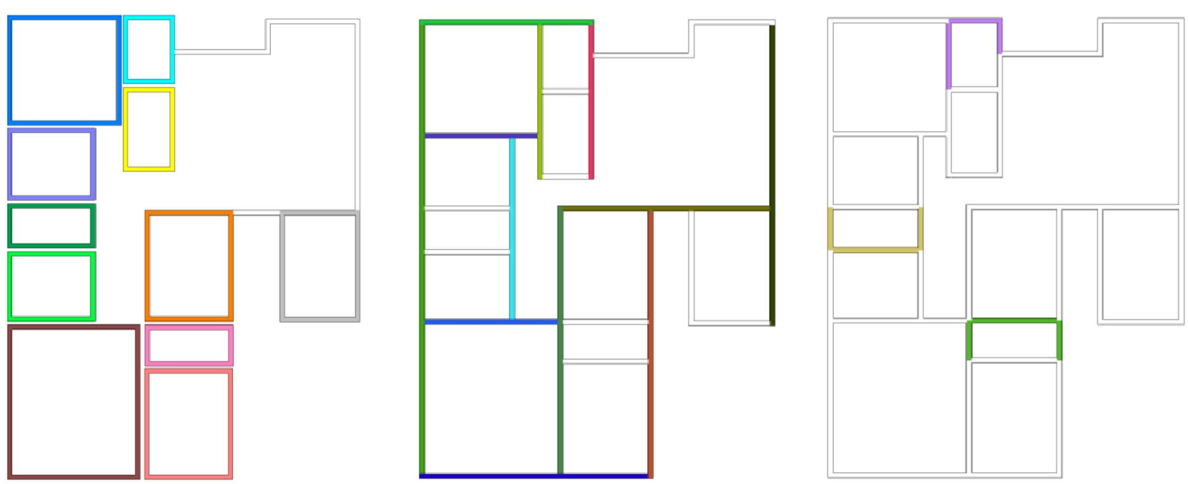

Fig. 18. (a) Fully volumetric solution (11VM); (b) fully panelized solution (0VM); and (c) three-VM solution.

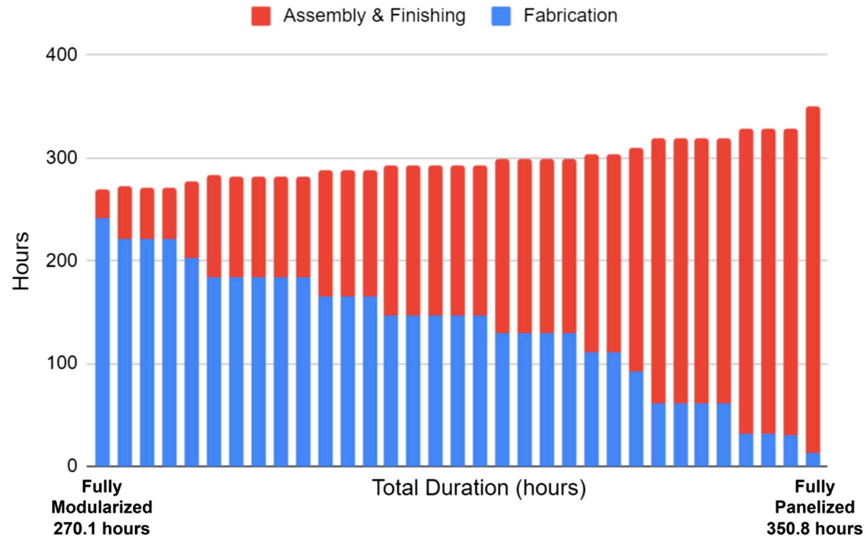

Fig. 19. Contribution of the off-site (fabrication) and onsite (assembly and finishing) work to the total duration of the Pareto optimal solutions.

modules. As such, the off-site fabrication efficiencies were lost with the decrease of volumetric modularization, which resulted in an increase of the total duration.

On the other hand, varying trends were realized when observing the change of fabrication, finishing, assembly, shipping, and total cost over the modularization strategy spectrum, as shown in Fig. 20. For this illustration project case, changing from a fully panelized strategy to a fully volumetric strategy resulted in (1) a 62% increase in the fabrication cost due to the duplication of walls between the connected modules, (2) an 18% savings in finishing cost due to the realized efficiencies of performing this work in a controlled off-site environment, (3) around a 10-fold increase in the assembly cost due to the increased rigging and connection time, and (4) around a threefold increase in the shipping cost due to the increased trucking trips of volumetric modules. As such, the increases in fabrication, assembly, and shipping costs outweighed the cost savings of the finishing work and resulted in increased costs for the increased level of volumetric modularization.

Sensitivity Analysis

A sensitivity analysis was performed to further validate the proposed methodology by assessing the sensitivity of the time and cost evaluations to changes of the input parameters. Sensitivity analysis is useful to assess the influential model parameters and apportion the model's sources of uncertainties (Razavi and Gupta 2015). The analysis was designed to assess the variations of the input

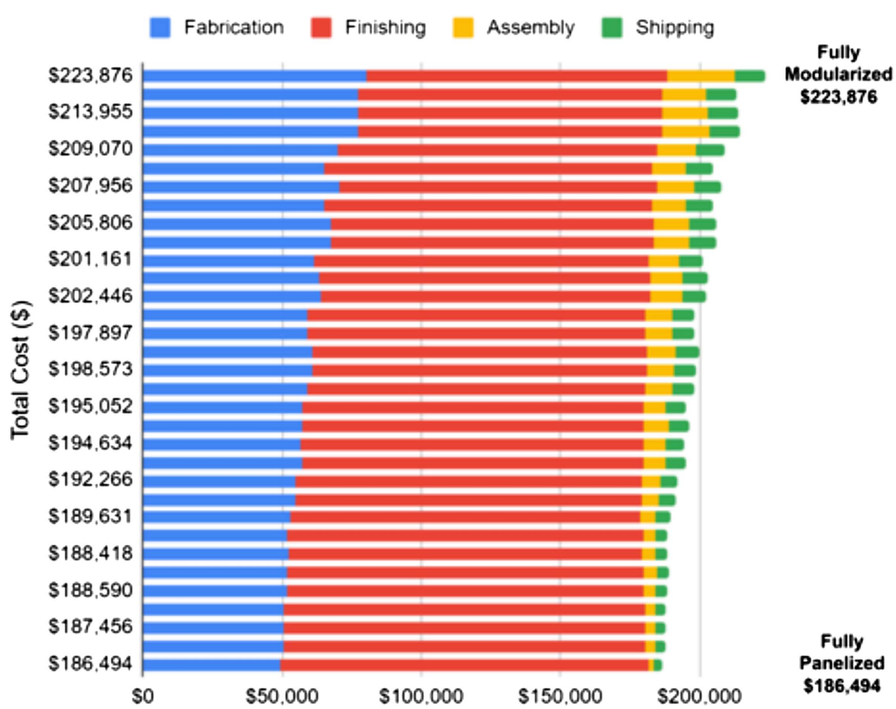

Fig. 20. Contribution of the fabrication, assembly, and shipping to the total cost of the Pareto optimal solutions.

parameters from their base values list in Table 2, which were changed one by one for two scenarios representing the upper and lower bounds of the value variation. A total of 15 input parameters were considered: seven time-related parameters (CONTm2m, CONTp2m, CONTp2p, PLT, CRTm, CRTp, and FT), seven cost-related parameters (FC, FLFC, MACR, PLC, TRC, TCP, and PACR), and the relative unit rate parameter (β). To enable a sufficient exploration of the model sensitivity, +50% and −50% variations were considered for the upper- and lower-bound scenarios, respectively, for each input parameter. In addition to the already analyzed base-case scenario, the methodology was applied for 30 additional scenarios for the upper- and lower-bound changes of the considered 15 input parameters. An exception was applied to the relative unit rate parameter (β), where the upper bound was set to its maximum value of 1, indicating that no production efficiency gains with off-site operations relative to onsite. The results were recorded, presenting a comprehensive understanding of how each parameter, when individually altered, influences the overall optimization outcome.

The time and cost evaluations were found to be sensitive to changes in the parameters related to the finish work, the floor panel fabrication and the off-site work relative unit rate. Fig. 21 shows the sensitivity spider plots for the minimal total duration (TD) and total cost (TC), where the horizontal axis lists the three input parameter cases (lower bound, base, and upper bound) and the vertical axis relates to the time and cost values when changing each individual parameter [Figs. 21(a and b), respectively].

First, the time evaluation was found to be sensitive to the changes in FT and the off-site relative unit rate (β). For all other variables, the range of the TD value change was 0.6% to 2.6%, which is not significant over the considered 50% variation bounds. The TD value was found to be positively correlated to FT because it increased by 47.8% at the upper-bound value of FT and decreased by 49.3% at the lower-bound case. A similar positive correlation was observed between TD and β, except a smaller TD change was observed at the upper-bound value (27.6%) due to the smaller positive variation considered for B (100% instead of $1.5 \times 75\%$).

Second, the cost evaluation was found to be sensitive to the changes in the finish cost (FC) and floor panel fabrication cost (FLFC) parameters. Similar to the impact of FT on TD, the variation of FC had a significant impact on the project TC compared with other parameters. TC dropped by 35.6% when using FC's lower-bound value, and increased by 34.8% when using its upper-bound value. FLFC had a very similar impact on TC when using its upper-bound value, but its lower-bound value resulted in a smaller drop in TC (−11.5%). When examining the detailed output of FLFC's lower-bound case, the generated fully volumetric solution had 10 VM units (compared with 11 VM units for the base and upper-bound cases). It appears that the lower-bound case of FLFC passed a breakeven point where merging the wall segments into larger PW units resulted in a more dominating solution (i.e., shorter duration time or lower cost).

Discussion

The example results affirmed that the proposed method is helpful for building construction managers to visualize and understand the trade-offs between construction time and cost with the varying modularization degrees and the amount of off-site work. A fully volumetric solution accelerates the construction schedule due to the time savings that are achieved from the off-site fabrication efficiencies. On the other hand, a fully volumetric solution increases the construction cost due to the need for more truck loads and the redundant material use for wall and floor panels. However, the observed time-cost trade-off may be limited to the analyzed example and the estimated time and cost parameters. A different floor plan with a complicated layout may increase the cost of panelized solutions due to the use of shorter panels, or reduce the cost of a fully volumetric solution by fabricating the VM units in a mobile site factory and skip the need for their costly shipping.

In previous studies, the application of DSM and clustering algorithms has been demonstrated in MEP systems and housing units, as explored by Samarasinghe et al. (2019), Suárez et al. (2023), and Isaac et al. (2016). However, these studies were limited to cluster modules by interdependencies, without considering desired topological structures in floor plan modularization. Our work aligns more closely with the domain explored by Said et al. (2017), Almashaqbeh and El-Rayes (2022), and Ghannad and Lee (2023), who specifically focused on floor plan optimization. Unlike their works, which were confined to the use of single typologies of modules, such as VM or PW exclusively, our approach could integrate the VM and PW to improve the design modularization process while considering the time and cost implications. This not only benefits the designers to utilize topologically standard modules to allow for floor plan design variety and flexibility but also enables the manufacturers to quantitatively evaluate the optimal hybrid use of volumetric and panelized modules in different projects.

The main contributions of this study are threefold. First, the new computational methodology provides a novel modularization approach that identifies the building modules considering their

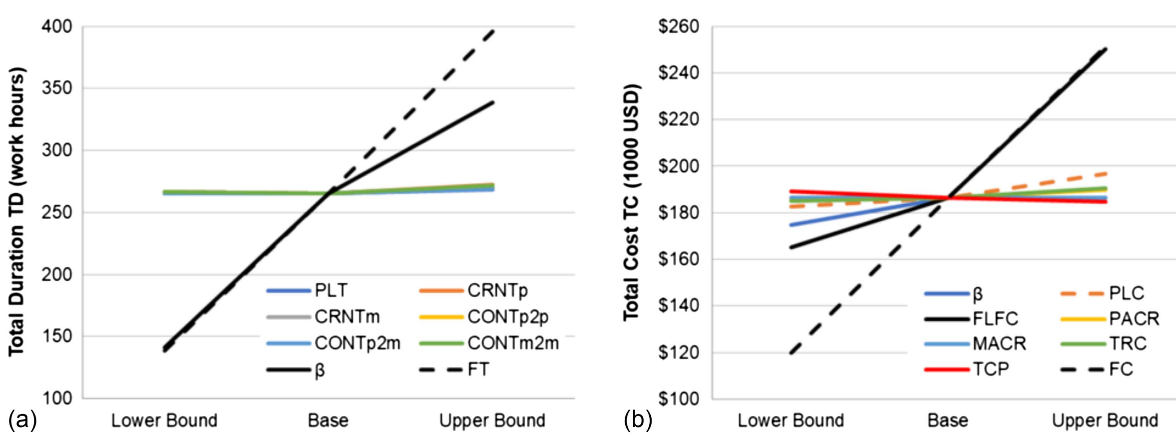

Fig. 21. Pareto front sensitivity to the changes of the input parameters: (a) total duration; and (b) total cost.

J. Constr. Eng. Manage., 2024, 150(8): 04024098

desired topological structures instead of their mere connectedness. Second, the presented methodology allows a balanced approach of modularization that utilizes topologically standard modules to allow for floor plan design variety and flexibility. Third, the developed time and cost metrics enable the manufacturers to quantitatively evaluate the optimal hybrid use of volumetric and panelized modules in different projects.

Research Limitations and Recommendations

The extensive enumeration of the volumetric modular construction benefits and barriers was not fully considered in this study, which may also alter the observed time-cost trade-offs or require the consideration of additional trade-off analysis dimensions. The formulation of the time and cost optimization functions did not consider some potential indirect benefits of fully volumetric solutions, including (1) economy of scale in manufacturing of multiple repeated VM units (Lawson et al. 2012), (2) the improved product quality of off-site fully fabricated and finished modules (Kamali and Hewage 2016; Lawson et al. 2012), (3) reduced safety issues due to better ergonomic conditions of off-site production and the reduce onsite work scope (Al-Hussein et al. 2009; Lawson et al. 2012), (4) the waste generation reduction that can be achieved from off-site manufacturing (Loizou et al. 2021; Quale et al. 2012), and (5) the reduced embodied energy and carbon footprint of off-site construction (Aye et al. 2012; Hammad et al. 2019). In terms of barriers, many studies have mentioned that VM requires a high initial investment to own and operate a modular factory (Ferdous et al. 2022; Liew et al. 2019; Pan et al. 2018). However, most aforementioned studies relied on industry reports or case studies. To support a computational modularization approach, more general quantifiable metrics are needed.

The topologies of panels, VM, and connections modeled in this study are not sufficient to represent the product variety in off-site construction. For instance, the topology of panels could be further categorized into (1) panels with openings and without openings, (2) sloped panels and flat panels, and (3) exterior panels and interior panels, ending up with different production times and costs (Said et al. 2017). Besides, this study only took angles as connection properties. Connections with specific design requirements (e.g., fire resistance, insulation, or structural capacity) can be explored and modeled in future studies by representing these attributes using node attributes and edge attributes.

This study assumed a one-to-one mapping relationship between rooms and VM. In other words, a room can be built using only one module, and a module does not include multiple rooms. One-to-many and many-to-one relationships between rooms and VM units could support a more flexible floor plan design. For instance, a large open space could be built with multiple open-sided VM units (one-to-many), whereas an entrance, a bathroom, and a living room could compose a studio VM (many-to-one).

Future work is needed to solve these limitations. First, product diversity, such as material type and geometry, should be considered in the module representation process. These features could be stored as node attributes in the graph models. Second, the one-to-many and many-to-one relationships between rooms and VM units could be encoded as graph rewriting rules. Graph rewriting rules concern the technique of creating a new graph out of an original graph algorithmically (Courcelle 1990). For example, a large open space as the input graph node could be translated to multiple output nodes representing the combination of the VMs.

Third, apart from the production time and cost, other project performance regarding modular construction could be explored, such as disassembly complexity. A floor plan design with varying degrees of modularization might result in a different disassembly process and amount of waste generation (Arisya and Suryantini 2021). Multiobjective optimization could be used to generate designs that meet the project performance objectives.

Finally, a case-based study should be performed to assess the usefulness and practicality of the proposed methodology by implementing it in an actual modular building project and collecting structured feedback from the project's manufacturer and team. Stochastic or fuzzy modeling can be implemented to expand the proposed methodology to consider probabilistic or vague values of the input parameters.

Conclusions

This study proposed a graph-based hybrid modularization optimization methodology for building floor plan designs. The floor plan and hybrid modules (panels and VM) were encoded via graph-based modeling. Building upon construction cost and time calculation, optimal module assignment plans were generated via genetic algorithms supported by graph operations. A Revit plug-in implementing the methodology was developed to support real-time interactive design and decision-making processes.

The approach solved three research gaps. First, previous studies did not take topological constraints into the modularization process. Topological constraints guarantee the structural integrity, transportation, and assembly feasibility of the generated design. Off-site construction uses two main products, panels and VM, which have more diversity, such as four-sided VM and partially open-sided VM. This study modeled the topological constraints by graphs, and the constraints were checked through the graph pattern-matching process. Assemblies that match the graph-based topologies were identified as potential VM or PW.

Second, by formalizing topological constraints early in the design process, this approach can generate modularized solutions that are not limited to a unique topology. More topologies of VM would achieve a higher level of design flexibility and better optimization results. Third, this study integrated both panels and VM to fit the design requirements. Given a floor plan, the generated solutions range from a fully panelized design to a fully volumetric solution. Decision makers are supported to balance the cost-time trade-off in an interactive design environment. This trade-off was examined in the illustrative example, where a fully volumetric solution was 23% faster than the fully panelized solution but was 22% more expensive.

In conclusion, the present study automated and optimized the process of how a customized design could be produced using a mass production approach during the design development stage. It supports the decision-making process of off-site construction projects at the early design phase while encouraging further research in this field.

The findings of this study can prove useful for modular and off-site building manufacturers to improve their agility and increase their market share. This study provides a novel computational methodology to explore the optimal balanced utilization of volumetric and panelized prefabrication approaches. Manufacturers do not need to be confined by a single prefabrication approach that limits their portfolio to a limited scope of the project (i.e., the panelized approach) or a limited industry segment (i.e., the volumetric approach). As such, the manufacturer with hybrid modularization operations will be more agile to satisfy the unique design requirements of different market segments and individual projects.

Data Availability Statement

Some or all data, models, or code that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments

This work was supported by Swiss National Science Foundation (Grant No. 197294).

Notation

The following symbols are used in this paper:

$A(X)$ = area of element X (m^2);
BUF_1 = time buffer between the panel fabrication and their volumetric module fabrication (h);
BUF_2 = time buffer between the onsite panel assembly and their rooms' interior finishing (h);
$C_{assembly}$ = cost of assembling the panels and volumetric modules ($);
C_{fab} = cost of fabricating the panels and volumetric modules ($);
C_{finish} = cost of interior finishes done off-site and onsite ($);
C_{ship} = cost of shipping and handling of the panels and volumetric modules ($);
$CONT_X$ = time of connecting X (h/connection);
$CRNT_X$ = crane lifting time per X (h/element);
FC_X = finish cost of room or VM X ($/$m^2$);
FLFC = floor panel fabrication cost ($/$m^2$);
FT_X = finish time of room or VM X (h);
$L(X)$ = length of element X (m);
M or m = volumetric module (VM);
MACR = VM assembly crew cost rate ($/h);
MAT = module assembly time onsite;
MFT = volumetric module fabrication time that includes the wall assembly and interior finishes (h);
$m2m$ = VM to VM connection;
$N(X)$ = quantity of element X (count);
nr = nonvolumetric room;
P or p = panel;
P_{2D} = panel for nonvolumetric rooms;
P_{3D} = panel for VM;
P_m = panel for VM m;
PACR = panel assembly crew cost rate ($/h);
PAT = panel assembly time onsite (h);
PFT_1 = panel fabrication time for the wall panels that will be used in volumetric module fabrication (h);
PFT_2 = panel fabrication time for the wall panels that will be assembled on site for the panelized scope (h);
PLC = fabrication cost for a panel linear unit ($/m);
PLT = panel unit length fabrication time (h/m);
$p2m$ = panel-to-VM connection;
$p2p$ = panel-to-panel connection;
r = room;
SFT = site finish time, the total time to finish the interiors of panelized rooms (h);
STK_1 = inventory buffer of panels between their fabrication and their use in volumetric module fabrication (panels);
STK_2 = inventory buffer of panels between their onsite assembly and the interior finishes of their rooms (panels);
TAF = total assembly and finish time onsite (h);
TCP = truck capacity load of panels (panels/truck);
TFB = total fabrication time of prefabricated units (panelized and volumetric) (h);
TRC = trucking round-trip cost ($/truck trip); and
β = off-site relative unit rate factor (%).

References

Abdul Nabi, M., and I. H. El-adaway. 2020. "Modular construction: Determining decision-making factors and future research needs." *J. Manage. Eng.* 36 (6): 04020085. https://doi.org/10.1061/(ASCE)ME.1943-5479.0000859.

Ahmadi, R., T. A. Roemer, and R. H. Wang. 2001. "Structuring product development processes." *Eur. J. Oper. Res.* 130 (3): 539–558. https://doi.org/10.1016/S0377-2217(99)00412-9.

Aiello, G., G. La Scalia, and M. Enea. 2012. "A multi objective genetic algorithm for the facility layout problem based upon slicing structure encoding." *Expert Syst. Appl.* 39 (12): 10352–10358. https://doi.org/10.1016/j.eswa.2012.01.125.

Al-Hussein, M., J. D. Manrique, and D. Mah. 2009. *North Ridge* CO_2 *analysis report comparison between modular and on-site construction*. Edmonton, AB, Canada: Univ. of Alberta.

Almashaqbeh, M., and K. El-Rayes. 2021. "Optimizing the modularization of floor plans in modular construction projects." *J. Build. Eng.* 39 (Nov): 102316. https://doi.org/10.1016/j.jobe.2021.102316.

Almashaqbeh, M., and K. El-Rayes. 2022. "Multiobjective model for optimizing the planning of floor plans, finishing level, and transportation in modular construction." *J. Archit. Eng.* 28 (4): 04022029. https://doi.org/10.1061/(ASCE)AE.1943-5568.0000563.

Anvari, B., P. Angeloudis, and W. Y. Ochieng. 2016. "A multi-objective GA-based optimisation for holistic manufacturing, transportation and assembly of precast construction." *Autom. Constr.* 71 (Part 2): 226–241. https://doi.org/10.1016/j.autcon.2016.08.007.

Arisya, K. F., and R. Suryantini. 2021. "Modularity in design for disassembly (DfD): Exploring the strategy for a better sustainable architecture." *IOP Conf. Ser.: Earth Environ. Sci.* 738 (1): 012024. https://doi.org/10.1088/1755-1315/738/1/012024.

Attouri, E., Z. Lafhaj, L. Ducoulombier, and B. Linéatte. 2022. "The current use of industrialized construction techniques in France: Benefits, limits and future expectations." *Cleaner Eng. Technol.* 7 (Apr): 100436. https://doi.org/10.1016/j.clet.2022.100436.

Aye, L., T. Ngo, R. H. Crawford, R. Gammampila, and P. Mendis. 2012. "Life cycle greenhouse gas emissions and energy analysis of prefabricated reusable building modules." *Energy Build.* 47 (Mar): 159–168. https://doi.org/10.1016/j.enbuild.2011.11.049.

Ayinla, K., F. Cheung, and M. Skitmore. 2022. "Process waste analysis for offsite production methods for house construction: A case study of factory wall panel production." *J. Constr. Eng. Manage.* 148 (1): 05021011. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002219.

Baldwin, C. Y., and K. B. Clark. 2018. *What is modularity? Design rules*. Cambridge, MA: MIT Press.

Bianconi, F., M. Filippucci, and A. Buffi. 2019. "Automated design and modeling for mass-customized housing: A web-based design space catalog for timber structures." *Autom. Constr.* 103 (May): 13–25. https://doi.org/10.1016/j.autcon.2019.03.002.

Blank, J., and K. Deb. 2020a. "A running performance metric and termination criterion for evaluating evolutionary multi- and many-objective optimization algorithms." In *Proc., 2020 IEEE Congress on Evolutionary Computation, CEC 2020–Conf. Proc.*, 2020003. New York: IEEE.

Blank, J., and K. Deb. 2020b. "Pymoo: Multi-objective optimization in python." *IEEE Access* 8 (May): 89497–89509. https://doi.org/10.1109/ACCESS.2020.2990567.

Boafo, F. E., J. H. Kim, and J. T. Kim. 2016. "Performance of modular prefabricated architecture: Case study-based review and future pathways." *Sustainability* 8 (6): 558. https://doi.org/10.3390/su8060558.

Bonnici, V., R. Giugno, A. Pulvirenti, D. Shasha, and A. Ferro. 2013. "A subgraph isomorphism algorithm and its application to biochemical data." *BMC Bioinf.* 14 (Apr): S13. https://doi.org/10.1186/1471-2105-14-S7-S13.

Boothroyd, G. 1994. "Product design for manufacture and assembly." *Comput. Aided Des.* 26 (7): 505–520. https://doi.org/10.1016/0010-4485(94)90082-5.

Cao, J., D. F. Bucher, D. M. Hall, and M. Eggers. 2022. "A graph-based approach for module library development in industrialized construction." *Comput. Ind.* 139 (Aug): 103659. https://doi.org/10.1016/j.compind.2022.103659.

Cao, J., D. F. Bucher, D. M. Hall, and J. Lessing. 2021. "Cross-phase product configurator for modular buildings using kit-of-parts." *Autom. Constr.* 123 (Mar): 103437. https://doi.org/10.1016/j.autcon.2020.103437.

Courcelle, B. 1990. "Graph rewriting: An algebraic and logic approach." In *Formal models and semantics*. Amsterdam, Netherlands: Elsevier.

Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan. 2002. "A fast and elitist multiobjective genetic algorithm: NSGA-II." *IEEE Trans. Evol. Comput.* 6 (2): 182–197. https://doi.org/10.1109/4235.996017.

Duncheva, T., and F. F. Bradley. 2019. "Multifaceted productivity comparison of off-site timber manufacturing strategies in mainland Europe and the United Kingdom." *J. Constr. Eng. Manage.* 145 (8): 04019043. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001641.

Duray, R., P. T. Ward, G. W. Milligan, and W. L. Berry. 2000. "Approaches to mass customization: Configurations and empirical validation." *J. Oper. Manage.* 18 (6): 605–625. https://doi.org/10.1016/S0272-6963(00)00043-7.

Eppinger, S. D., and T. R. Browning. 2018. "Introduction to design structure matrix methods." *Des. Struct. Matrix Methods Appl.* 1 (Mar): 1–15. https://doi.org/10.7551/mitpress/8896.003.0003.

Feist, S., L. Sanhudo, V. Esteves, M. Pires, and A. A. Costa. 2022. "Semi-supervised clustering for architectural modularisation." *Buildings* 12 (3): 303. https://doi.org/10.3390/buildings12030303.

Ferdous, W., A. Manalo, A. Sharda, Y. Bai, T. D. Ngo, and P. Mendis. 2022. "Construction industry transformation through modular methods." In *Innovation in construction*, 259–276. New York: Springer.

Forsythe, P. J., and S. M. E. Sepasgozar. 2019. "Measuring installation productivity in prefabricated timber construction." *Eng. Constr. Archit. Manage.* 26 (4): 578–598. https://doi.org/10.1108/ECAM-09-2017-0205.

Ghannad, P., and Y.-C. Lee. 2023. "Optimizing modularization of residential housing designs for rapid postdisaster mass production of housing." *J. Constr. Eng. Manage.* 149 (7): 04023046. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002390.

Golberg, D. E. 1989. *Genetic algorithms in search optimization & machine learning*, 412. Boston: Addison-Wesley Longman.

Gosling, J., M. Pero, M. Schoenwitz, D. Towill, and R. Cigolini. 2016. "Defining and categorizing modules in building projects: An international perspective." *J. Constr. Eng. Manage.* 142 (11): 1–11. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001181.

Hagberg, A., P. Swart, and D. S. Chult. 2008. *Exploring network structure, dynamics, and function using NetworkX*. Los Alamos, NM: Los Alamos National Laboratory.

Hammad, A. W., A. Akbarnezhad, P. Wu, X. Wang, and A. Haddad. 2019. "Building information modelling-based framework to contrast conventional and modular construction methods through selected sustainability factors." *J. Cleaner Prod.* 228 (Mar): 1264–1281. https://doi.org/10.1016/j.jclepro.2019.04.150.

Isaac, S., T. Bock, and Y. Stoliar. 2016. "A methodology for the optimal modularization of building design." *Autom. Constr.* 65 (May): 116–124. https://doi.org/10.1016/j.autcon.2015.12.017.

Jiao, J., T. W. Simpson, and Z. Siddique. 2007. "Product family design and platform-based product development: A state-of-the-art review." *J. Intell. Manuf.* 18 (1): 5–29. https://doi.org/10.1007/s10845-007-0003-2.

Kamali, M., and K. Hewage. 2016. "Life cycle performance of modular buildings: A critical review." *Renewable Sustainable Energy Rev.* 62 (Sep): 1171–1183. https://doi.org/10.1016/j.rser.2016.05.031.

Khalili, A., and D. K. Chua. 2014. "Integrated prefabrication configuration and component grouping for resource optimization of precast production." *J. Constr. Eng. Manage.* 140 (2): 04013052. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000798.

Lawson, M., R. Ogden, and C. Goodier. 2014. Design in modular construction. In *Design in modular construction*. Boca Raton, FL: CRC Press.

Lawson, R. M., and R. G. Ogden. 2008. "'Hybrid' light steel panel and modular systems." *Thin Walled Struct.* 46 (7–9): 720–730. https://doi.org/10.1016/j.tws.2008.01.042.

Lawson, R. M., R. G. Ogden, and R. Bergin. 2012. "Application of modular construction in high-rise buildings." *J. Archit. Eng.* 18 (2): 148–154. https://doi.org/10.1061/(ASCE)AE.1943-5568.0000057.

Liew, J. Y. R., Y. S. Chua, and Z. Dai. 2019. "Steel concrete composite systems for modular construction of high-rise buildings." *Structures* 21 (Oct): 135–149. https://doi.org/10.1016/j.istruc.2019.02.010.

Loizou, L., K. Barati, X. Shen, and B. Li. 2021. "Quantifying advantages of modular construction: Waste generation." *Buildings* 11 (12): 622. https://doi.org/10.3390/buildings11120622.

Lucko, G., and E. M. Rojas. 2012. "Research validation in the construction domain." In *Construction research congress 2009: Building a sustainable future*, 1449–1458. Reston, VA: ASCE.

McGraw-Hill Construction. 2011. "Prefabrication and modularization: Increasing productivity in the construction industry." In *SmartMarket report*. New York: McGraw-Hill Construction.

Mehlhorn, K., and D. Michail. 2009. "Minimum cycle bases: Faster and simpler." *ACM Trans. Algorithms* 6 (1): 1–13. https://doi.org/10.1145/1644015.1644023.

Nauata, N., K.-H. Chang, C.-Y. Cheng, G. Mori, and Y. Furukawa. 2020. "House-GAN: Relational generative adversarial networks for graph-constrained house layout generation." In *Proc., European Conf. on Computer Vision*, 162–177. New York: Springer.

Pan, W., Y. Yang, and L. Yang. 2018. "High-rise modular building: Ten-year journey and future development." In *Proc., Construction Research Congress 2018*, 523–532. Reston, VA: ASCE. https://doi.org/10.1061/9780784481301.052.

Pizarro, P. N., N. Hitschfeld, I. Sipiran, and J. M. Saavedra. 2022. "Automatic floor plan analysis and recognition." *Autom. Constr.* 140 (Dec): 104348. https://doi.org/10.1016/j.autcon.2022.104348.

Quale, J., M. J. Eckelman, K. W. Williams, G. Sloditskie, and J. B. Zimmerman. 2012. "Construction matters: Comparing environmental impacts of building modular and conventional homes in the United States." *J. Ind. Ecol.* 16 (2): 243–253. https://doi.org/10.1111/j.1530-9290.2011.00424.x.

Razavi, S., and H. V. Gupta. 2015. "What do we mean by sensitivity analysis? The need for comprehensive characterization of 'global' sensitivity in Earth and Environmental systems models." *Water Resour. Res.* 51 (5): 3070–3092. https://doi.org/10.1002/2014WR016527.

RSMeans. 2018. *Building construction costs with RSMeans data*. Edited by S. C. Plotner. Rockland, MA: Gordian.

RSMeans. 2022. *Square foot costs with RSMeans data*. Rockland, MA: Gordian.

Said, H., and K. El-Rayes. 2013. "Performance of global optimization models for dynamic site layout planning of construction projects." *J. Autom. Constr.* 36 (1): 71–78. https://doi.org/10.1016/j.autcon.2013.08.008.

Said, H. M., T. Chalasani, and S. Logan. 2017. "Exterior prefabricated panelized walls platform optimization." *Autom. Constr.* 76 (Apr): 1–13. https://doi.org/10.1016/j.autcon.2017.01.002.

Salama, T., A. Salah, O. Moselhi, and M. Al-Hussein. 2017. "Near optimum selection of module configuration for efficient modular construction." *Autom. Constr.* 83 (Jun): 316–329. https://doi.org/10.1016/j.autcon.2017.03.008.

Salvador, F., C. Forza, and M. Rungtusanatham. 2002. "Modularity, product variety, production volume, and component sourcing: Theorizing beyond generic prescriptions." *J. Oper. Manage.* 20 (5): 549–575. https://doi.org/10.1016/S0272-6963(02)00027-X.

Samarasinghe, T., T. Gunawardena, P. Mendis, M. Sofi, and L. Aye. 2019. "Dependency structure matrix and hierarchical clustering based algorithm for optimum module identification in MEP systems." *Autom. Constr.* 104 (May): 153–178. https://doi.org/10.1016/j.autcon.2019.03.021.

Schmidt, R., K. S. Vibaek, and S. Austin. 2014. "Evaluating the adaptability of an industrialized building using dependency structure matrices."

Construct. Manage. Econ. 32 (1–2): 160–182. https://doi.org/10.1080/01446193.2013.847274.
Sharafi, P., B. Samali, H. Ronagh, and M. Ghodrat. 2017. "Automated spatial design of multi-story modular buildings using a unified matrix method." *Autom. Constr.* 82 (Feb): 31–42. https://doi.org/10.1016/j.autcon.2017.06.025.
Sharma, D., and C. Chattopadhyay. 2018. "High-level feature aggregation for fine-grained architectural floor plan retrieval." *IET Comput. Vision* 12 (5): 702–709. https://doi.org/10.1049/iet-cvi.2017.0581.
Shewchuk, J. P., and C. Guo. 2012. "Panel stacking, panel sequencing, and stack locating in residential construction: Lean approach." *J. Constr. Eng. Manage.* 138 (9): 1006–1016. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000520.
Sinha, K., S. Y. Han, and E. S. Suh. 2020. "Design structure matrix-based modularization approach for complex systems with multiple design constraints." *Syst. Eng.* 23 (2): 211–220. https://doi.org/10.1002/sys.21518.
Ślusarczyk, G. 2018. "Graph-based representation of design properties in creating building floorplans." *Comput. Aided Des.* 95 (Feb): 24–39. https://doi.org/10.1016/j.cad.2017.09.004.
Smith, R. E. 2010. *Prefab architecture: A guide to modular design and construction*. New York: Wiley.
Stone, R. B., K. L. Wood, and R. H. Crawford. 2000. "A heuristic method for identifying modules for product architectures." *Des. Stud.* 21 (1): 5–31. https://doi.org/10.1016/S0142-694X(99)00003-4.
Suárez, J. L., L. Gosselin, and N. Lehoux. 2023. "Optimizing modularity of prefabricated residential plumbing systems for construction in remote communities." *J. Constr. Eng. Manage.* 149 (1): 05022017. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002393.
Tidhar, N., Y. Li, K. S. Kristombu Baduge, P. Mendis, P. S. M. Thilakarathna, and E. R. K. Chandrathilaka. 2021. "A Slack approach to optimised modularisation of prefabricated structures using a multi-variable modularisation index." *Structures* 33 (May): 1235–1251. https://doi.org/10.1016/j.istruc.2021.04.064.
Tserng, H. P., Y. L. Yin, E. J. Jaselskis, W. C. Hung, and Y. C. Lin. 2011. "Modularization and assembly algorithm for efficient MEP construction." *Autom. Constr.* 20 (7): 837–863. https://doi.org/10.1016/j.autcon.2011.03.002.
Ulrich, K. T., and S. D. Eppinger. 2018. "Product design and development." In *Handbook of research on new product development*. 6th ed. New York: McGraw-Hill Education.
Vala, T. M., V. N. Rajput, Z. W. Geem, K. S. Pandya, and S. C. Vora. 2021. "Revisiting the performance of evolutionary algorithms." *Expert Syst. Appl.* 175 (Aug): 114819. https://doi.org/10.1016/j.eswa.2021.114819.
Van Beek, T. J., M. S. Erden, and T. Tomiyama. 2010. "Modular design of mechatronic systems with function modeling." *Mechatronics* 20 (8): 850–863. https://doi.org/10.1016/j.mechatronics.2010.02.002.
Wang, B., B. P. Gilbert, H. Guan, and L. H. Teh. 2016. "Shape optimisation of manufacturable and usable cold-formed steel singly-symmetric and open columns." *Thin Walled Struct.* 109 (Oct): 271–284. https://doi.org/10.1016/j.tws.2016.10.004.
Wang, X., Y. Yang, and K. Zhang. 2018. "Customization and generation of floor plans based on graph transformations." *Autom. Constr.* 94 (Dec): 405–416. https://doi.org/10.1016/j.autcon.2018.07.017.
Wee, T. P. Y., M. Aurisicchio, and I. Starzyk. 2017. "Evaluating modularisation tools in construction." In *Proc., 34th Int. Symp. on Automation and Robotics in Construction*. Oulu, Finland: International Association for Automation & Robotics in Construction.
Xu, Z., T. Zayed, and Y. Niu. 2020. "Comparative analysis of modular construction practices in mainland China, Hong Kong and Singapore." *J. Cleaner Prod.* 245 (Feb): 118861. https://doi.org/10.1016/j.jclepro.2019.118861.
Yang, Z., Z. Ma, and S. Wu. 2016. "Optimized flowshop scheduling of multiple production lines for precast production." *Autom. Constr.* 72 (Dec): 321–329. https://doi.org/10.1016/j.autcon.2016.08.021.
Yazdi, A. J., A. A. F. Fini, and P. Forsythe. 2021. "Mass-customisation of cross-laminated timber wall systems at early design stages." *Autom. Constr.* 132 (Aug): 103938. https://doi.org/10.1016/j.autcon.2021.103938.
Yu, T. L., A. A. Yassine, and D. E. Goldberg. 2007. "An information theoretic method for developing modular architectures using genetic algorithms." *Res. Eng. Des.* 18 (2): 91–109. https://doi.org/10.1007/s00163-007-0030-1.