buildings

MDPI

Article

Automated Prefabricated Slab Splitting Design Using a Multipopulation Coevolutionary Algorithm and BIM

Chengran Xu [1], Xiaolei Zheng [2], Zhou Wu [2] and Chao Zhang [3,*]

[1] College of Civil Engineering, Zhejiang University of Technology, Hangzhou 310024, China; chengran-xu@outlook.com

[2] School of Automation, Chongqing University, Chongqing 400044, China; xiaolei-zheng@stu.cqu.edu.cn (X.Z.); zhouwu@cqu.edu.cn (Z.W.)

[3] School of Civil Engineering, Chongqing University, Chongqing 400044, China

[*] Correspondence: superzhang1028851150@gmail.com

Abstract: The prefabricated composite slab (PCS) is an essential horizontal component in a building, which is made of a precast part and a cast-in-place concrete layer. In practice, the floor should be split into many small PCSs for the convenience of manufacturing and installation. Currently, the splitting design of PCS mostly relies on sound knowledge and valuable experience of construction. While rule-based parametric design tools using building information modeling (BIM) can facilitate PCS splitting, the generated solution is suboptimal and limited. This paper presents an intelligent BIM-based framework to automatically complete the splitting design of PCSs. A collaborative optimization model is formulated to minimize the composite costs of manufacturing and installation. Individuals with similar area information are grouped into a subpopulation, and the optimization objective is to minimize the specifications and quantities of PCSs. Through the correlation information within the subpopulation and the shared information among each other, the variable correlation is eliminated to accomplish the task of collaborative optimization. The multipopulation coevolution particle swarm optimization (PSO) algorithm is implemented for the collaborative optimization model to determine the sizes and positions of all PCSs. The proposed framework is applied in the optimized splitting design of PCSs in a standard floor to demonstrate its practicability and efficiency.

Keywords: prefabricated composite slabs; splitting design; building information modeling; multipopulation coevolutionary algorithm; collaborative optimization

check for updates

Citation: Xu, C.; Zheng, X.; Wu, Z.; Zhang, C. Automated Prefabricated Slab Splitting Design Using a Multipopulation Coevolutionary Algorithm and BIM. *Buildings* **2024**, *14*, 433. https://doi.org/10.3390/buildings14020433

Academic Editor: Giuseppina Uva

Received: 31 December 2023
Revised: 26 January 2024
Accepted: 1 February 2024
Published: 5 February 2024

1. Introduction

The prefabricated composite slab (PCS) is an important horizontal component of buildings, which is manufactured in a controlled environment and transported to the construction site [1,2]. Due to their ability to accelerate construction timelines, enhance manufacturing quality control, and reduce on-site labor and material, PCSs are widely utilized in construction [3,4]. In practice, the sizes of prefabricated components are limited by convenience and controllability in manufacturing, transportation, and installation processes. The PCS is usually not designed as a single large element to directly replace the traditional cast-in-place slab. Hence, the splitting design of PCSs is intended to strategically plan the floor space to be filled with smaller precast elements and cast-in-place concrete areas. In a real project, given the floor spaces for a prefabricated building, the dimensions and positions of hundreds of slabs need to be determined with the sound knowledge of construction and valuable experience from an engineer. Layouts of PCSs are provided in two-dimensional (2D) construction drawing to guide the manufacturing and assembly process. However, it is essential to comprehensively consider the requirements of logistics, manufacturing, transportation, installation, and structural aspects in the splitting design to enhance efficiency and reduce the costs of a construction project [5]. Hence, the optimized design of PCS splitting is a multiobjective problem. Even with the aid of a computer, a splitting design only relies on simple rules and still requires tedious labor due to the complex requirements in a series of construction processes, implying more proneness to human errors. Hence, the need to provide an automated splitting design of PCSs to improve the construction efficiency is warranted.

In the architecture, engineering, and construction (AEC) industry, a splitting design is usually completed using building information modeling (BIM) software (Revit2019) for automatic PCS layout. Using a few specialized tools in BIM software, the floors are split into multiple smaller slabs according to predefined rules and parameter settings. With the aid of BIM technology, the sizes and locations of PCSs for a building can be better three-dimensionally (3D) represented [6]. While rule-based splitting design can satisfy fundamental structural performance and functional requirements, the provided results are not necessarily the optimal solution. The increased quantity and specification of PCSs not only decrease production efficiency but also incur additional costs for transportation and installation processes. With the escalation of demands for quality and budget control in construction projects, the splitting design of PCSs becomes increasingly complex and variable. Furthermore, the slab size and layout should be flexibly adjusted to eliminate inappropriate joints and avoid interacting with pipelines and preserved holes. BIM-based parametric design aims to enable designers to efficiently manipulate adaptive designs [7]. While parameter design tools improve the efficiency of repetitive tasks, the creative and customized design aspects are limited, thereby increasing the time and effort required for frequent design changes. Hence, an intelligent BIM-based design tool is essential to facilitate the splitting design of PCSs to improve design quality and efficiency.

In recent years, many scholars have conducted research to complete the parametric design of buildings with the aid of BIM. Banihashemi et al. [8] combined parametric design with modular coordination to reduce construction material waste using the Rhinoceros 3D Grasshopper platform. Zou et al. [9] developed a parameterized design approach to complete the arrangement and virtual simulation of building machines. The building machine was parametrically modeled using the visual programming tool Geometry Nodes in Blender open-source software (2019), and the Unity3D program (2019) was utilized to visualize the virtual scene. Wang et al. [10] utilized an improved genetic algorithm to deal with the assembly sequence optimization of prefabricated concrete buildings, and BIM was used to establish the parametric modeling and visualize the assembly process of prefabricated component. To improve the constructability of the prefabricated building design, some researchers incorporate the concept of design for manufacture and assembly (DfMA) to coordinate construction requirements [11–14]. Yuan et al. [11] proposed a DfMA-oriented parametric method for prefabricated building design. Bakhshi et al. [14] introduced an efficient framework of combining the BIM and DfMA approach with mass customization to assist clients in taking part in the offshore construction configuration procedure. In the splitting of prefabricated components, Feng et al. [15] developed design software using the BIM platform to realize the splitting and detailing of PCS in the design phase. The production data could be exported to bridge the design and manufacturing process. Dong et al. [16] presented a DFMA-oriented modular design approach applied to vertical components in a building, aiming to improve the standardization of component specifications. The mold utilization rates and production efficiency were increased due to the standardized manufacture and installation of components. Xiao and Bhola [17] utilized the parametric design platform Dynamo and BIM software Revit to achieve the prefabricated component splitting according to predefined rules based on the design and construction requirements. Although existing research can achieve a splitting design based on parameter settings and predefine rules, it is difficult to obtain an optimal solution. Moreover, parametric design relying on BIM software is time-consuming, laborious, and inefficient, encountering large and various projects. Therefore, the development of an intelligent optimization approach is necessary for the large-scale PCS splitting problem.

In conclusion, the existing literature has some key limitations in solving the splitting design of prefabricated components. Firstly, the parametric design method based on BIM is time-consuming and results in complicated engineering, posing challenges in meeting the requirements of a creative and customized design. Secondly, it is difficult to obtain an ideal solution with a method based on parameter settings and predefined rules. In the splitting design of PCSs, the sizes of PCSs should be optimized to minimize the number and specification of PCSs, which can reduce the costs and time of manufacturing and installation. Generally, a high-rise prefabricated building has a significant number of PCSs, especially if the owner and contractors of the building project emphasize the efficiency and speed of construction. Hence, the PCS splitting is a large-scale multiobjective optimization (MOO) problem with multiple constraint conditions and interrelated variables. Most of the traditional methods for solving large-scale MOO problems are based on mathematical techniques, Pareto techniques, and other implementations [18]. Population-based intelligent optimization algorithms have no special limitation on objective functions and constraint conditions and have features such as adaptive search capability. Hence, population-based algorithms have been widely used in the optimization process of MOO problems [19]. Zhang et al. [20] proposed a multiobjective collaborative particle swarm optimization (PSO) algorithm, aiming to achieve an energy-saving optimization of multitrains in terms of stopping time, departure interval, and train speed. Lu et al. [21] proposed a two-stage PSO algorithm based on population cooperation to solve large-scale MOO problems and enhance the ability of subpopulation collaboration and overall convergence speed. Xu et al. [22] proposed a cooperative evolution algorithm to efficiently address MOO problems through grouping correlated decision variables. Wang et al. [23] proposed an adaptive differential evolution (DE) mechanism based on coevolution and covariance to reduce the dimensionality and complexity of the problem, considering the correlation between variables. Ma et al. [24] proposed a multiobjective evolutionary algorithm based on decision variable analysis. The interdependent and control variable analysis were introduced to deal with the high-dimensionality issues of the optimization problem. Du et al. [25] proposed a multiobjective energy management strategy based on the PSO algorithm to simultaneously reduce vehicle energy consumption and battery over-discharge. Fan et al. [26] implemented a multiobjective and multiparameter thermodynamic optimal design of a liquid air energy storage system, employing the PSO algorithm to achieve optimal system performance. The PSO algorithm has been widely used and researched in various fields due to its properties such as a fast convergence, few parameters, simplicity of implementation, and flexibility [27]. Inspired by the above research, this paper adopts a multipopulation coevolution optimization PSO algorithm to solve the high-dimensional splitting design problem of PCSs.

Based on the multipopulation coevolution approach, an intelligent BIM-based framework is proposed to complete the splitting design of PCSs. The framework combines the structural design information and construction requirements for a prefabricated building to optimize the layouts of PCSs. The outlines and dimensions of each floor, as well as the subareas utilizing PCSs in the whole building are extracted from structural plan drawings and BIM models. Based on the structural design information, an intelligent PCS splitting module using a collaboration optimization is developed to minimize the composite costs of manufacturing and installation. The sizes of all PCSs are considered as variables in the optimization process. Due to a large number of PCSs required in a building, the splitting design is a large-scale discrete combination optimization problem. A multipopulation coevolution PSO algorithm is introduced to implement the splitting design module. The global optimization problem can be decomposed into serval solvable low-dimensional subproblems to improve the computational efficiency according to variable decomposition and correlation analysis. The multipopulation approach can effectively solve the dimensionality curse, and the dimension of the search space in a subpopulation is largely decreased to find the reliable optimal solution. Furthermore, the BIM models of all PCSs compliant with the IFC (Industry Foundation Classes) standard are all generated automatically. The IFC standard is the most common neutral file format for data exchange between heterogeneous software. The PCS layout drawing is exported from the IFC model to provide the size and location of all PCSs.

This paper is organized as follows. In Section 2, the construction requirements of the PCS layout and the mathematical model of the splitting design optimization are described. Section 3 proposes an intelligent BIM-based framework for the PCS splitting. In Section 4, an intelligent PCS layout method is formulated based on the collaboration optimization and the developed multipopulation coevolution approach for the proposed framework is described. Section 5 presents illustrative examples to demonstrate the efficiency and performance of the proposed framework. Conclusions are included in Section 6.

2. Problem Description

2.1. PCS Layouts in Prefabricated Building

As shown in Figure 1, a prefabricated composite slab (PCS) consists of a precast composite layer and a cast-in-place layer. PCSs are supported by beams and walls and are categorized into unidirectional slabs and bidirectional slabs considering their force characteristics and support conditions. According to the construction requirements of a four-side supported slab in the concrete structure design code [28], if the ratio of the long side to the short side exceeds or is equal to three, the slabs should be designed according to the unidirectional slab design; if the ratio of the long side to the short side is less than three, the slabs are designed according to the bidirectional slab design.

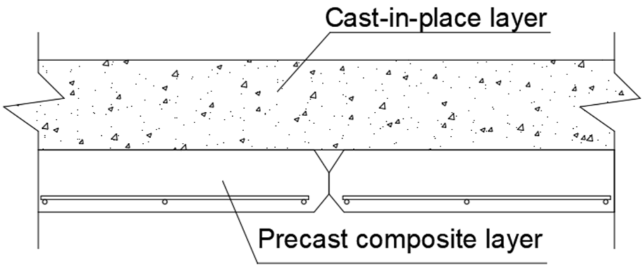


Figure 1. Composition of prefabricated composite slab.

As shown in Figure 2, separate joints (Figure 3a) are utilized between unidirectional slabs, which can be assembled at any position. The integrated joints (Figure 3b) are used between bidirectional slabs. According to the standard drawing set of a truss-reinforced concrete composite slab [29], the joint width of a bidirectional slab is 300 mm, and the unidirectional slabs adopt a width of zero between them. As shown in Tables 1 and 2, the nominal width of a PCS (Figure 4) is the sum of the actual slab width and the joint width. The nominal width of both the unidirectional and bidirectional slabs is between 1200 mm and 2400 mm, with a total of five grades. As shown in Table 3, the span of a bidirectional slab is between 3000 mm and 6000 mm, increasing by 300 mm between each grade. As shown in Figure 5, a PCS can be categorized into an edge slab and a middle slab according to its arrangement position, and the two types of slabs with the same nominal width have different actual widths. As shown in Table 4, the nominal span is the sum of the actual span and the joints' width, and the span of a unidirectional slab is between 2700 mm and 4200 mm with an increase of 300 mm between each grade.

The splitting design needs to consider the constraints of transportation, production, assembly, and layout areas of PCSs, which should not exceed 6 m in length and 3 m in width. Due to the edge distance requirements of the truss rebar, the width of the remaining slab in all splitting areas should be no less than 400 mm and should be maximized for mechanical properties. In the detailing design process, engineers rely on design specifications and a priori knowledge to split the floors into different sizes of PCSs. Due to the complexity of the actual project, it is difficult to obtain the optimal solution in a manual way, and the PCS splitting optimization design needs to be investigated.

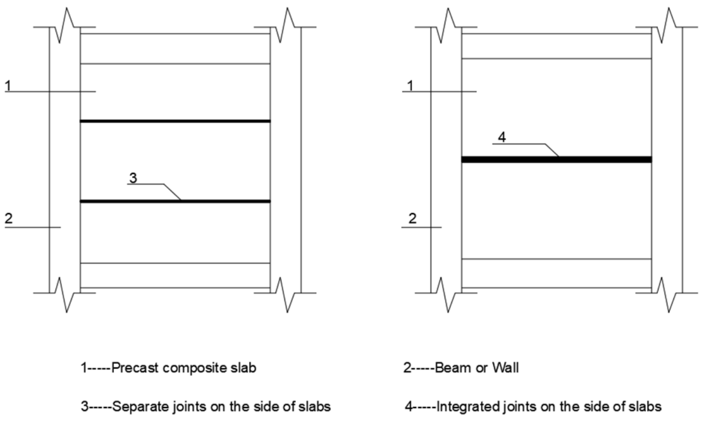
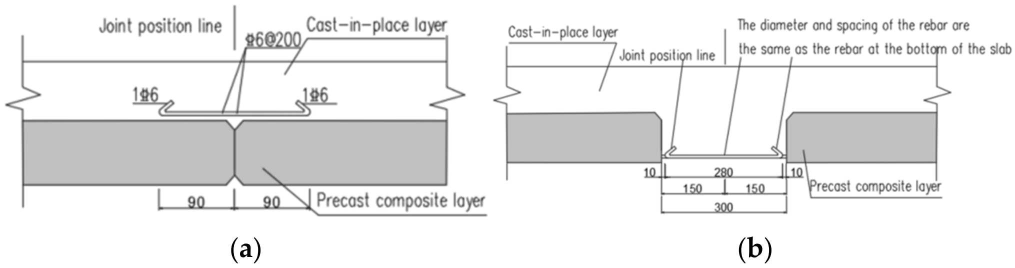

1
2
3
1
4
2
1-----Precast composite slab
2-----Beam or Wall
3-----Separate joints on the side of slabs
4-----Integrated joints on the side of slabs

Figure 2. Joint type on the side of PCSs.



(**a**) (**b**)

Figure 3. Joint construction: (**a**) unidirectional slab; (**b**) bidirectional slab.

Table 1. Width of a bidirectional slab.

Grade	W12	W15	W18	W20	W24
Nominal width (mm)	1200	1500	1800	2000	2400
Actual width of edge slab (mm)	960	1260	1560	1760	2160
Actual width of middle slab (mm)	900	1200	1500	1700	2100

Table 2. Width of a unidirectional slab.

Grade	W12	W15	W18	W20	W24
Nominal width (mm)	1200	1500	1800	2000	2400
Actual width (mm)	1200	1500	1800	2000	2400

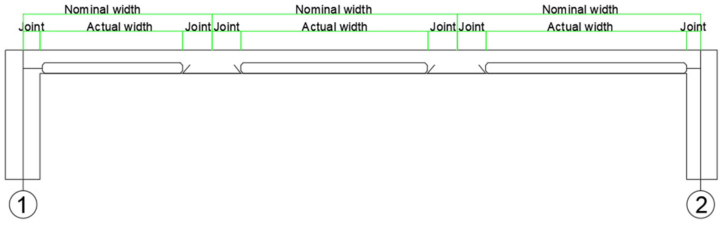
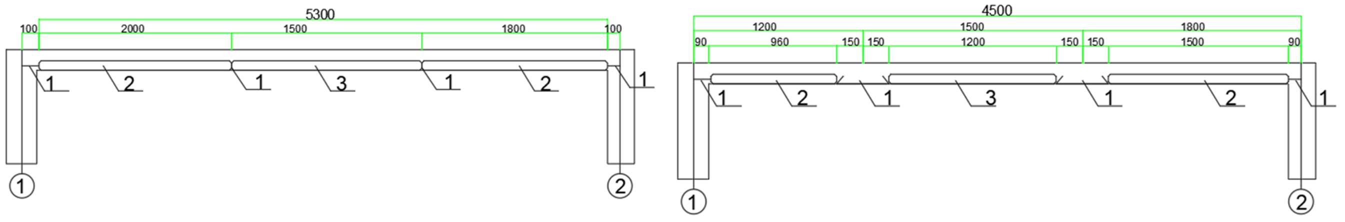


Figure 4. Diagram of nominal and actual widths.

Table 3. Span of a bidirectional slab.

Grade	**DL30**	**DL33**	**DL36**	**DL39**	**DL42**	**DL45**
Nominal span (mm)	3000	3300	3600	3900	4200	4500
Actual span (mm)	2820	3120	3420	3720	4020	4320
Grade	**DL48**	**DL51**	**DL54**	**DL57**	**DL60**	
Nominal span (mm)	4800	5100	5400	5700	6000	
Actual span (mm)	4620	4920	5220	5520	5820	



(**a**) (**b**)

Figure 5. Diagram of position relationship of the joints (unit: mm): (**a**). unidirectional slab; (**b**). bidirectional slab (1—joint; 2—edge slab; 3—middle slab).

Table 4. Span of a unidirectional slab.

Grade	**SL27**	**SL30**	**SL33**	**SL36**	**DL39**	**DL42**
Nominal span (mm)	2700	3000	3300	3600	3900	4200
Actual span (mm)	2520	2820	3120	3420	3720	4020

2.2. Splitting Design Optimization of PCSs

In the manufacturing process, PCSs with the same width or the same span grade can reduce the quantities of mold types, such as the whole mold, partial end mold, and partial side mold. In order to accurately describe the splitting design of PCS, a mathematical model is proposed. In terms of the specification quantities of the PCSs, an optimization model is developed based on the number of PCS specifications with the same slab span grade. The objective function is given by Equation (1)

$$\min R_{total}^{k} = \sum_{g=1}^{g_{sum}} w_g R_k^g \tag{1}$$

where R_{total}^{k} is the specification quantities of PCS, g_{sum} is the number of slab span grades, R_k^g is the span grade of the gth PCS, w_g is the weight coefficient of the span grade width types of the gth slab; the sum of the weight coefficients is one.

In terms of the number of precast composite slabs, an optimization model aims to obtain the minimum number of slabs, given by Equation (2).

$$\min R_{total}^{n} = \sum_{f=1}^{f_{sum}} R_n^f \tag{2}$$

where R_{total}^{n} is the number of PCSs, f_{sum} is the number of separating subareas, R_n^f is the number of PCSs arranged in fth subareas.

In this study, a set of optimization strategies is proposed for the correlation analysis of areas for arranging PCS. There are four types of correlation problems for separating areas, such as equal length and same grade width, equal length and different grade width, unequal length and same grade width, and unequal length and different grade width. The number (1, 2, 3) stands for different separating areas in the Figure 6. As shown in Figure 6a, there are two separation subareas with different sizes (4000 mm × 3700 mm, 4000 mm × 3900 mm). Considering the design, the manufacture requirements of two areas, the optimal PCS splitting solution is that the first area can be split into three PCSs with slab widths of 1800 mm, 1800 mm, and 400 mm and a slab span of 3700 mm, and the second area can be split into three PCSs with slab widths of 1800 mm, 1800 mm, 400 mm and a slab span of 3900 mm. Thus, the solution can reduce the specifications of PCSs and share the end and side molds. As shown in Figure 6b, there are three PCSs with different separation areas (3700 mm × 2200 mm, 3700 mm × 2900 mm, 3700 mm × 3200 mm). The optimal split solution is three separation subareas that can be split into three PCSs with slab widths of 1500 mm, 1500 mm, 700 mm, and slab spans of 2200 mm, 2900 mm, 3200 mm, respectively. Thus, these strategies can reduce the specifications of slabs to realize the sharing of the end mold. As shown in Figure 6c, there are two precast floor slabs with different lengths and widths in the same span grade (3500 mm × 3100 mm, 4000 mm × 3200 mm). The optimal split strategies are two separation areas that can be split into PCSs with widths of 1500 mm, 2000 mm, 500 mm, and different spans of 3100 mm and 3200 mm, respectively. Thus, these strategies can reduce the specifications of slabs to realize the sharing of the end and side molds. As shown in Figure 6d, there are two precast floor slabs with different lengths and widths in the different span grades (4600 mm × 1500 mm, 3600 mm × 2500 mm). The optimal split solution is two separation subareas that can be split into three PCSs with widths of 1800 mm, 1800 mm, 1000 mm, and different spans of 1500 mm and 2500 mm, respectively. Thus, these solutions can reduce the specifications of PCS to realize the sharing of the end mold.

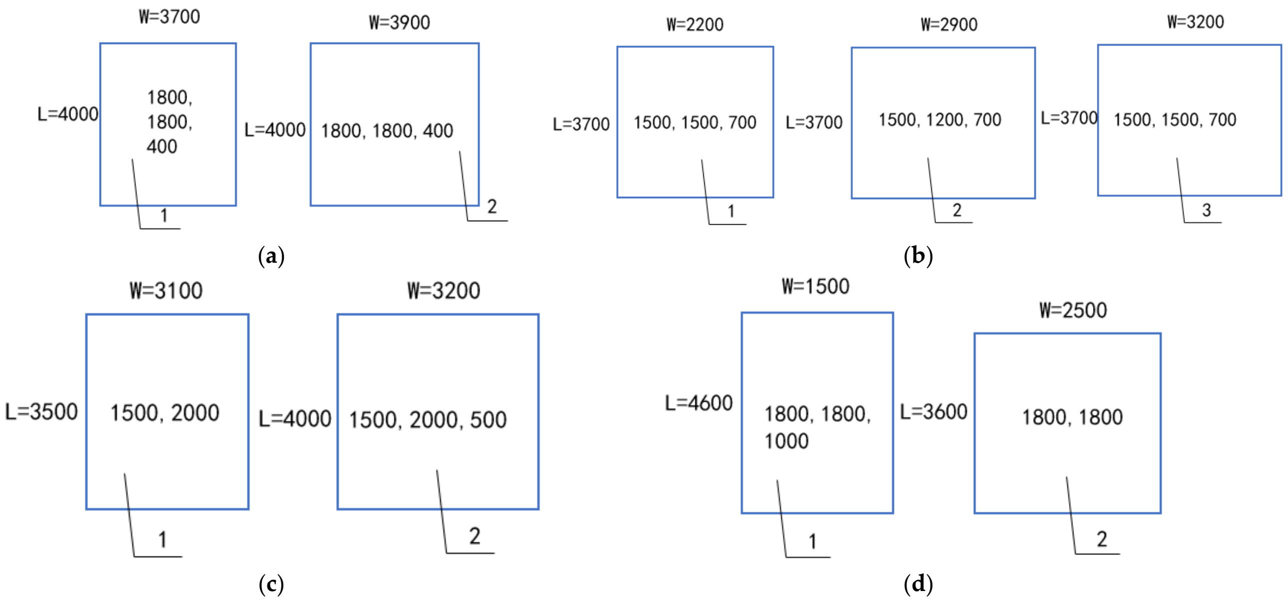


Figure 6. Four types of correlation analysis (unit: mm): (**a**) the strategy of example one; (**b**) the strategy of example two; (**c**) the strategy of example three; (**d**) the strategy of example four.

3. BIM-Based Framework for Splitting Design of PCSs

To improve the flexibility and variability of an automated PCS layout, an intelligent BIM-based framework for the splitting design is proposed to generate the 3D visualization and 2D drawing providing the sizes and locations of PCSs. As shown in Figure 7, the proposed framework consists of three modules: (1) a data input module, (2) a PCS splitting module, and (3) a data output module, discussed in the following.

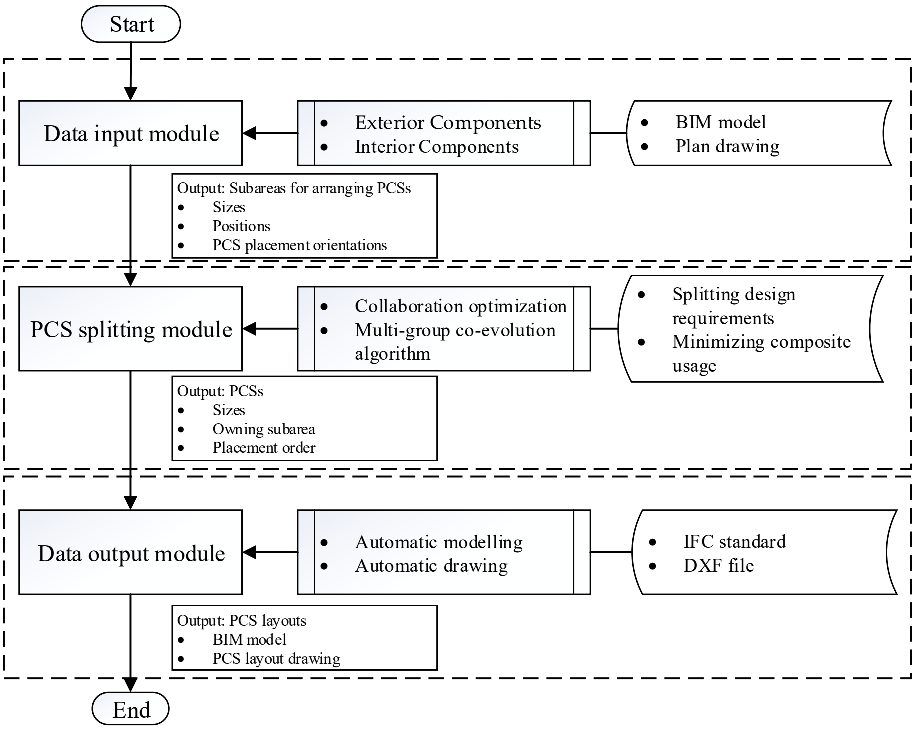


Figure 7. BIM-based framework for the splitting design of PCSs.

3.1. Data Input Module

The data input module aims to extract the necessary data to perform the splitting design of PCS in the next module. When the design of a prefabricated building enters the detailing stage, the structural plan drawing or BIM model should be provided. The outlines and dimensions of each floor in a building can be extracted to determine the entire slab area, based on the exterior structural components such as beams and columns. The entire slab area can be further divided into several rectangular subareas that require a PCS arrangement according to interior components such as beams and walls.

In the structural BIM of a building, the positions and dimensions of components can be easily obtained to create the splitting scenario of PCSs. The corner columns and the exterior beams connecting corner columns are taken as vertices and boundaries, respectively, to form the plan outline for each floor. The position and length of beams and walls in the interior of the floor are extracted to determine the dividing lines. The enclosed subareas are formed by the combination of boundary and interior lines and the size and position of each subarea can be generated. The subareas where PCSs can be arranged are automatically determined based on the construction requirements of prefabricated structures. Users can also specify the subareas where to arrange PCSs to enhance design flexibility. Moreover, the placement orientations of PCSs in each subarea can be specified by users or predefined rules, in which a PCS is always arranged along the shorter side of the subarea. In the structural plan drawing, the components are represented by graphical entities like lines, arcs, circles, etc. DXF (Drawing Exchange Format) is a file format for the exchange of CAD (computer-aided design) drawings. Graphical entities and information are often organized into layers to group related elements together [30]. The DXF files of a structural plan drawing are read and interpreted to obtain the layouts of exterior and interior components and determine the subareas for PCS arrangement. Hence, the structural plan drawing or BIM model of a building is prepared and imported in this module to extract the size, position, and PCS placement orientations used in the next module.

3.2. PCS Splitting Module

The PCS splitting module was developed to facilitate the intelligent optimization of the PCS layout. The information extracted from the structural BIM model or plan drawing for a building can be considered as the input data used in this module. By selecting the number and specifications of standard PCSs and adjusting the widths of nonstandard PCSs, the PCSs' splitting can be optimized to decrease the costs and enhance efficiency in the manufacturing, transportation, and installation processes.

To address the dimensionality curse problem in global optimization, a collaborative optimization model for the splitting design of PCSs in large and complex scenarios was formulated to provide the size, ownership, and placement order of each PCS. Combined with the information on subareas provided in the previous module, each PCS, considering its size, position, and placement orientation, can be provided to prepare for the convenient generation of the BIM model and construction drawing. The proposed optimization model is described in detail in Section 4.1.

3.3. Data Output Module

According to the optimization results of the PCS splitting module, the BIM model and construction drawing including the PCS layouts can be exported to the data output module. As a primary data exchange standard for BIM, the IFC standard is widely used in the information sharing among different software tools. The IFC model, which includes the geometric and semantic information of building components, is intended to facilitate interoperability in the AEC industries. It is a neutral and open standard that is not controlled by a single contractor or group of contractors [31]. Based on the IFC standard, the BIM models of PCSs are represented by entity IfcSlab. The size of each PCS obtained in the precious module are used to create the geometry entity IfcProductDefinitionShape through the entity IfcExtrucdedAreaSolid. In the IFC standard, components are located using a relative coordinate system. The related position of each PCS on a floor can be obtained according to its placement order and the position and PCS placement orientation of the owning subarea. The entity IfcLocalPlacement is used to create the local placement of the spatial structure such as the site, building, story, and components. The local placement of a PCS is created by the entity IfcLocalPlacement based on the position related to the story. The entity IfcRelContainedInSpatialStructure is used to assign a PCS to a certain story. All PCSs with customized sizes can be repetitively created and orientated toward their respective positions. An IFC file of PCSs is automatically generated to improve the modeling efficiency.

To generate the PCS layout drawing, the IFC file of PCSs is imported to parse the IfcExtrucdedAreaSolid and IfcLocalPlacement of each IfcSlab. The length, width, and position of PCSs are obtained to generate a 2D drawing. DWG and DXF are two different file formats used in AutoCAD software (2019) [32]. DWG is the native file format of AutoCAD, containing complete design data, including graphics, text, and metadata. DXF is an open file format used for exchanging data between different CAD programs, typically containing graphics and basic object information but not including metadata and other advanced features. Since DXF is open and can be interoperable between different CAD software programs, it was chosen as the file format for the PCS layout drawings. According to the geometry and position information, the PCS layouts are drawn in the DXF file of the structural plan drawing. In the DXF file, each PCS is repetitively drawn by several lines which are represented by the entity 'LINE' and the starting and ending points of each line are defined to specify the entity properties. The DXF file of the PCS layout drawing is automatically exported to reduce the drawing time in the splitting design stage.

4. Multipopulation Coevolution Optimization of PCS Splitting Design

The use of population-based intelligent optimization algorithms to solve the splitting design problem of PCSs needs to consider factors such as multiple constraints, multiple objectives, the large scale, and the variables in the problem. The traditional global optimization strategy treats all the decision variables in the problem as the element information of the individuals in population. The number of decision variables determines the individual's dimension. In the splitting design of PCSs, the spatial scale is large, and the number of PCSs is in the dozens or even hundreds. The problem of correlation between variables needs to be considered, which largely leads to a failure of the global optimization strategy due to the high dimensionality of the problem. Even if the number of iterations is increased to find a feasible solution, it may not be optimal for the problem. Therefore, a collaborative optimization model based on a multipopulation coevolution PSO algorithm is proposed to solve the high-dimensional PCS splitting design problem.

4.1. Collaborative Optimization Model

The objective of the PCS splitting design problem is to minimize the specifications and quantities of standard and nonstandard PCSs, given all the subareas for which PCSs need to be arranged. Certain subareas exhibit similar lengths and widths, and the optimization results demonstrate a significant correlation with the specifications and quantities of PCSs. Therefore, the subareas with identical or similar size information were integrated to reduce the dimensionality of the optimization problem. The subarea integration was achieved through the following steps: (1) grouping subareas with identical length and width in a set of solutions, (2) associating subareas with similar width to minimize the specifications and quantities of standard PCSs, and (3) merging subareas with the same length to decrease the diversity of PCS widths. In this context, a collaborative optimization model featuring multiple clusters was introduced to address the splitting design challenge of PCSs.

A multipopulation collaborative optimization algorithm aims to simulates the collaborative behavior of multiple populations in nature [33]. It is based on the principle of population intelligence and collaborative evolution theory in order to find the global optimal solution through cooperation among multiple subpopulations. The algorithm involves initializing multiple subpopulations, with each subpopulation independently searching for the local optimal solution within the solution space using its own search strategy. Simultaneously, it encourages cooperation between different subpopulations through information sharing and communication. Throughout the optimization process, the subpopulations continuously adapt their strategies and share experiences to enhance the global search capability. This algorithm demonstrates a strong robustness and global search capability in solving complex optimization problems, effectively addressing challenges such as multimodality and high dimensionality.

In a typical optimization problem with d-dimensional decision variables, an individual in the population is represented by $X = [x_1, x_2, x_3, \ldots, x_{n-1}, x_n]$. In the splitting design of PCSs, each x_i represents the layout scheme of the ith subarea, each having a certain dimension. To reduce the dimensionality of the original optimization problem, variables with higher correlation are combined and treated as a subpopulation for the joint optimization, while variables with no correlation form a separate subpopulation. The original high-dimensional MOO problem is decomposed into several low-dimensional subproblems for resolution. As shown in Figure 8, a population with size n is taken as an example in the global optimization strategy, and the individuals with d-dimensional decision variables are decomposed into m groups of low-dimensional subpopulations with size n, which effectively improves the optimization success rate and efficiency.

The subpopulation cooperative mechanism is used to solve the original MOO problem and the above subgroup decomposition can reduce the dimensionality of the solution. In the splitting design of PCSs, the regions with variable correlation can be combined as a subpopulation. In the optimization process, subpopulations of certain subareas find a local optimal solution with a high variable correlation to decrease the quality of the optimization results of the whole population. The optimization effect between subpopulations is synergized through information sharing. Therefore, a cooperative approach is utilized for decomposing the original populations to exchange and share information within the subpopulations, as well as between different subpopulations, preventing the issues of falling into a local optimum. Figure 9 gives the structure of the multipopulation cooperative optimization. Firstly, the optimal solution sequences of each subpopulation are obtained by the cooperative optimization approach; then, the subpopulations are gradually merged based on the information sharing, and the combination of the optimal solution sequences is corrected; finally, the optimal solution sequence of the original MOO problem is obtained by integrating the solution sequences of all subpopulations.

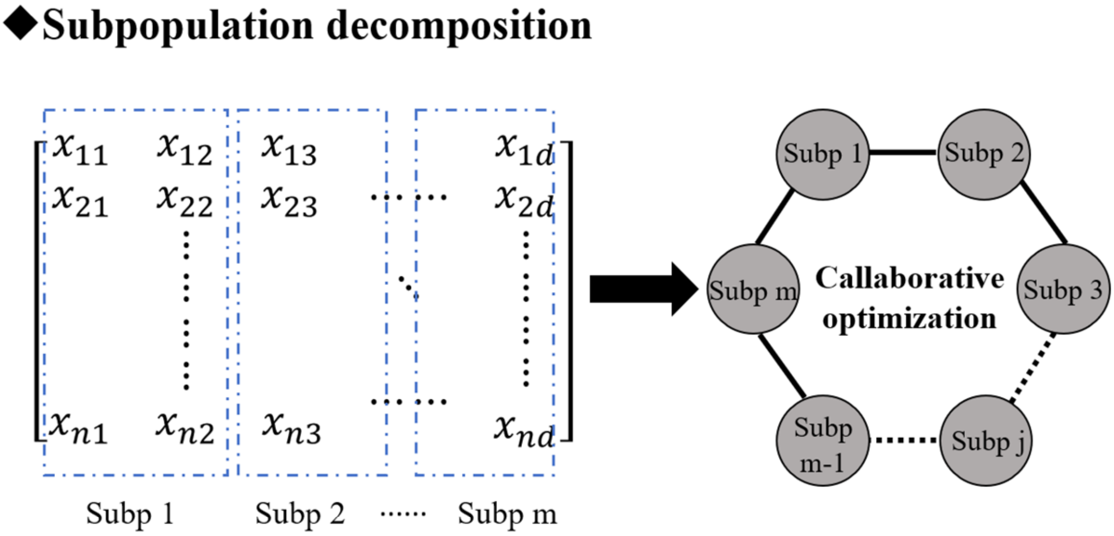

◆Subpopulation decomposition
x_{11} x_{12} x_{13} x_{1d}
x_{21} x_{22} x_{23} x_{2d}
x_{n1} x_{n2} x_{n3} x_{nd}
Subp 1
Subp 2
Subp m
Subp 1
Subp 2
Subp 3
Subp j
Subp m-1
Subp m
Callaborative optimization

Figure 8. Subpopulation decomposition for high-dimensional variables.

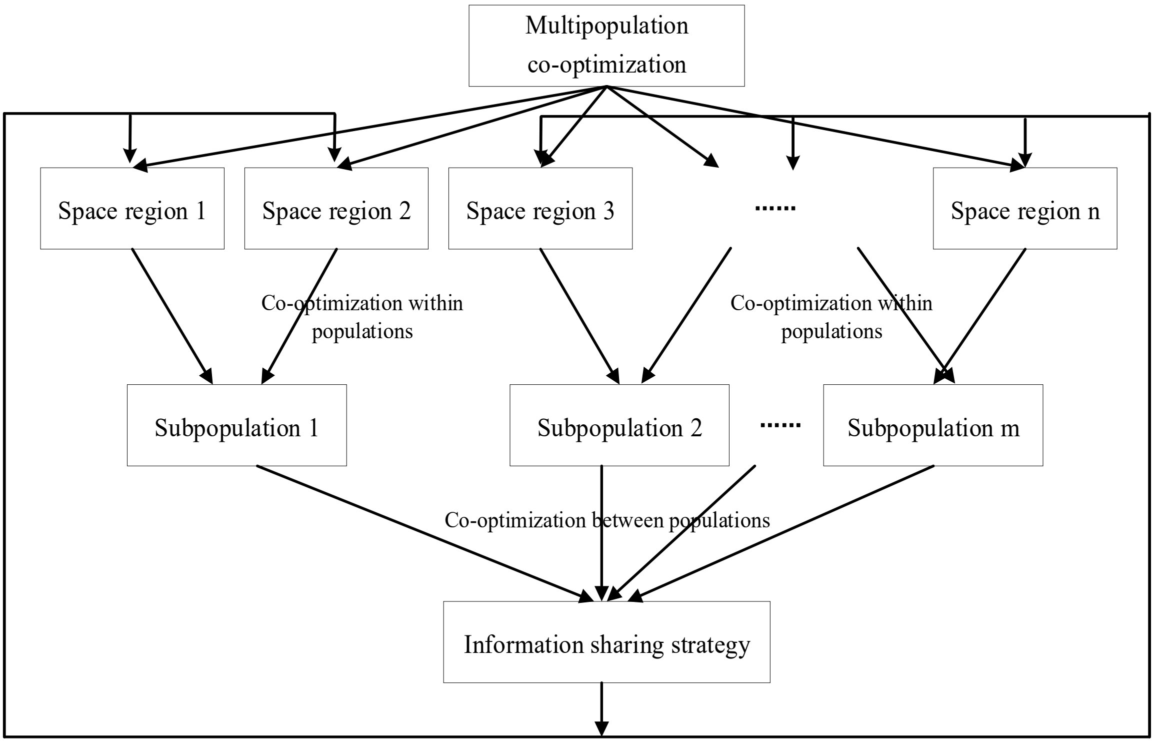

Multipopulation co-optimization
Space region 1
Space region 2
Space region 3
......
Space region n
Co-optimization within populations
Co-optimization within populations
Subpopulation 1
Subpopulation 2
......
Subpopulation m
Co-optimization between populations
Information sharing strategy

Figure 9. Structure for the multipopulation collaborative optimization.

In the PCS splitting optimization of subareas, the optimization objectives are the specifications and quantities of PCS widths, which can be expressed as follows

$$f_{best}^{l}(x_i):\ \min R_{total}^{k}, R_{total}^{n} \tag{3}$$

where $f^{l}_{best}(x_i)$ denotes the optimal solution in subarea i.

To obtain the best combined optimal solution after synergy, the optimal solution relationships between each subpopulation need to be comprehensively considered. Therefore, the found optimal solution for each subpopulation is saved and the individual with the best synergistic effect is selected to accelerate the optimization efficiency of the collaborative optimization process. The subpopulation synergistic optimization model can be expressed as

$$f^{subi}_{best}(x_{subi}) : \text{ min } R^{k}_{total}, R^{n}_{total}, E^{k}_{total}\left(\text{option}\left[f^{l\_list}_{best}(x_{subi\_1}), \ldots, f^{l\_list}_{best}\left(x_{subi\_j}\right)\right]\right. \tag{4}$$

where $f^{subi}_{best}(x_{subi})$ denotes the combined optimal solution for the ith subpopulation; the width of the nonstandard PCS E^{k}_{total} is included as one of the objective functions due to the necessity of regional correlation analysis; and $f^{l\_list}_{best}\left(x_{subi\_j}\right)$ represents the selected optimal solutions in subarea j. Collaborative optimization within the subpopulation facilitates the discovery of a set of relatively superior solutions.

Similarly, through information sharing among subpopulations, the collaborative optimization is achieved to obtain a set of better solutions for the whole population. Therefore, the optimization model for the whole population can be represented as:

$$f^{g}_{best}(X) : \text{ min } R^{k}_{total}, R^{n}_{total}, E^{k}_{total}\left(\text{option}\left[f^{sub1\_list}_{best}(x_{sub1}), \ldots, f^{subm\_list}_{best}(x_{subi})\right]\right. \tag{5}$$

where $f^{g}_{best}(X)$ denotes the global optimization objective, and $f^{subi\_list}_{best}(x_{subi})$ denotes the optimal solution of the ith subpopulation. The collaborative optimization among subpopulations can effectively increase the solution diversity for the PCS splitting design problem.

In summary, the splitting design problem of PCSs can be summarized as (1) grouping the subareas based on the length and width information, and decomposing the original high-dimensional problem into m low-dimensional subproblems for solving; (2) collaboratively optimizing the objectives within each subpopulation to obtain a set of optimal solution sequences; and (3) realizing the collaborative optimization of multiple subpopulations based on the information sharing among them to finally obtain the optimal solution sequence of the original problem.

4.2. Multipopulation Coevolution Algorithm PSO

The splitting design problem of PCS entails a vast search space dimension, leading to an exponentially increasing optimization time under the global optimization strategy. The optimization process may even face failure in obtaining a better solution. In this regard, the multipopulation collaborative optimization model is introduced to address the original multiobjective optimization problem. By grouping the variables of the original problem, the solution dimension is significantly reduced. Simultaneously, the correlation issues between subpopulations are resolved in the collaborative optimization process to derive a set of optimal solution sequences for the original problem. Considering the advantages of the PSO algorithm in solving multiobjective optimization problems, the optimization framework of the PSO algorithm based on multipopulation coevolution was incorporated into the optimization process of each subpopulation.

The idea of the PSO algorithm originated from the study of the foraging behavior of bird flocks, where the flocks share information collectively to enable the group to find the optimal destination [34]. In the PSO algorithm, the entire population is referred to as a flock and the individuals are referred to as particles. Each particle has two attributes, i.e., position and velocity. The optimal position searched by each particle is denoted by $pbest_i = \left[pbest_{i,1}, pbest_{i,2}, \ldots, pbest_{i,D}\right]$, and the optimal position searched by the population is denoted by $gbest = [gbest_1, gbest_2, \ldots, gbest_D]$. During each iteration, the particles update their velocity and position as

$$v_{i,j} = \omega \cdot v_{i,j} + lr_1 \cdot rand \cdot \left(pbest_{i,j} - x_{i,j}\right) + lr_2 \cdot rand \cdot \left(gbest_j - x_{i,j}\right) \tag{6}$$

$$x_{i,j} = x_{i,j} + v_{i,j} \tag{7}$$

where j $(j = 1, 2, \ldots, D)$ is the index of dimension; $v_i = (v_{i,1}, v_{i,2}, \ldots, v_{i,D})$ is the velocity vector of the ith particle; and $x_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,D})$ is the position vector of the ith particle. ω is the inertia weight providing a balance between exploration and exploitation. lr_1 and lr_2 are the positive learning rates, and *rand* is a random vector uniformly distributed in [0, 1].

In the PSO algorithm, the particle velocity is updated using three different parts. The specific explanation is as follows: (1) The inertia part, which consists of inertia weight ω and the particle's own velocity $v_{i,j}$, represents the particle's trust in the previous motion state. (2) The cognitive part represents the particle's individual cognition, i.e., the own experience of the particle, which can be understood as the distance and direction between the current position $x_{i,j}$ of the particle and its own historical optimal position $pbest_{i,j}$. (3) The social part indicates the information sharing and cooperation between particles, i.e., the experience derived from other excellent particles in the population, which can be understood as the distance and direction between the current position $x_{i,j}$ of the particle and the historical optimal position $gbest_j$ of the population. Based on the guidance of the above heuristic information, the particle can effectively learn from the population to update its velocity and further update its position. This search method can effectively speed up the convergence of the population.

The classical PSO algorithm is mostly employed to tackle continuous optimization problems. However, in the splitting design problem of PCSs, the standard PCS width, based on the design specification, constitutes a discrete set of values. Therefore, the variable part of the classical PSO algorithm was adapted to suit the discrete optimization process of the original problem. Utilizing the standard PCS widths outlined in the optimization problem (Section 2.1), the PCS width variables were encoded in integer form, as detailed in Table 5.

Table 5. Integer code representation of PCS width.

Value of Integer Coding	Actual Value of PCS Width (mm)	Value of Integer Coding	Actual Value of PCS Width (mm)
0	0	4	2000
1	1200	5	2400
2	1500	6	[400, 1200)
3	1800		

The feasible decision variables in the solution process are encoded in integer form, while the perturbation of the particle velocity is a continuous variable, making the updated position information a floating-point number. In this regard, the paper modifies the position information of individuals in the PSO algorithm as follows

$$x_{i,j} = \begin{cases} x_{i,j} & , \quad x_{i,j} = round\left(x_{i,j}\right) \\ randint(x\_min, x\_max)\ , & if \ \ x_{i,j} < x\_min \ or \ \ x_{i,j} > x\_max \end{cases} \tag{8}$$

where $round()$ is a rounding function, using rounding methods to round the individual; *x_min* and *x_max* are the lower and upper bounds of the individual x, respectively, which determine the range of the feasible decision space of the individual; and $randint()$ is a function that generates random integers within a given range.

The specific steps of the above optimization process for the PSO algorithm based on integer coding are as follows:

(1) Initialize all particles by assigning values to their velocities and positions. Set the historical optimal *pbest* of an individual as the current position, and the optimal individual in the population as the current *gbest*.

(2) Update the velocity and position of each dimension of each particle *i* according to Equations (6) and (7), respectively, and round based on Equation (8).

(3) In each generation of evolution, the fitness function value of each particle is calculated. If the current fitness function value is better than the historical optimum, update *pbest*. If the current fitness function value is better than the global historical optimum, update *gbest*.

(4) Determine if the termination condition (maximum number of iterations or accuracy requirements) has been met; if not, proceed to (2) to continue execution.

The pseudocode for PSO based on integer coding is given in Algorithm 1.

Algorithm 1: Pseudocode for PSO based on integer coding

```
Input: population size $NP$; dimension $D$; maximum generation $G_m$; learning rate
       $lr_1$, $lr_2$; inertia weight $\omega$; fitness function $f(x)$
Output: the best solution in the final population
$G \leftarrow 1$; Randomly choose $NP$ individuals $x$ with $D$ dimension; $v \leftarrow 0$; $pbest \leftarrow x$;
  $gbest \leftarrow x(1)$;
while $G < G_m$ do
    for $i \leftarrow 1$ to $NP$ do
        for $j \leftarrow 1$ to $D$ do
            $v_{i,j} = \omega \cdot v_{i,j} + lr_1 \cdot rand \cdot (pbest_{i,j} - x_{i,j}) + lr_2 \cdot rand \cdot (gbest_j - x_{i,j})$
            $x_{i,j} = x_{i,j} + v_{i,j}$ ; $x_{i,j} = \boldsymbol{round}(x_{i,j})$
            if $x_{i,j} < x\_min$ or $x_{i,j} > x\_max$ then
                $\boldsymbol{randint}(x\_min, x\_max)$
            end
        end
    end
    for $i \leftarrow 1$ to $NP$ do
        if $f(x_i) < f(pbest_i)$ then
            $pbest_i \leftarrow x_i$;
        end
        if $f(x_i) < f(gbest)$ then
            $gbest \leftarrow x_i$;
        end
    end
    $G \leftarrow G + 1$;
end
```

5. Experiment Result

5.1. Case Overview

As shown in Figure 10, a standard floor in a prefabricated building was used as the illustrative example for the PCS splitting design to verify the feasibility and efficiency of the proposed framework. The length and width of the standard floor were 284 m and

161 m, and the total area of the floor space was about 384.323 m^2. The structural plan drawing of the standard floor was imported into the data input module. The subareas for arranging PCSs were extracted to provide the related information such as sizes, positions, and PCS placement orientations, which are shown in Figure 10. In this standard floor, the quantity of subareas was 14 and the ranges of the length and width of the subareas were 3300 mm~10,450 mm and 3000 mm~3800 mm, respectively. As shown in Figure 10, the red and cyan colors express the prefabricated areas and the cast-in-place areas, respectively.

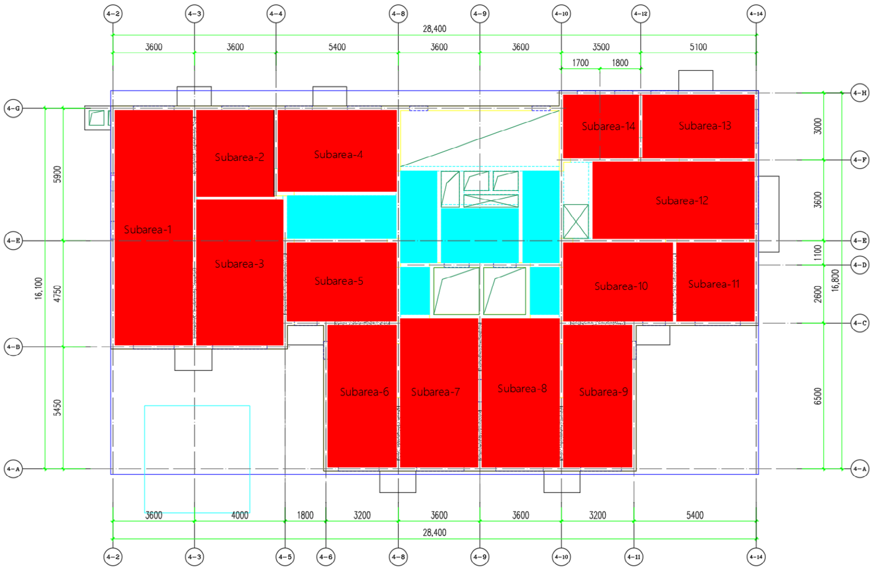


Figure 10. Plan drawing of a standard floor (unit: mm).

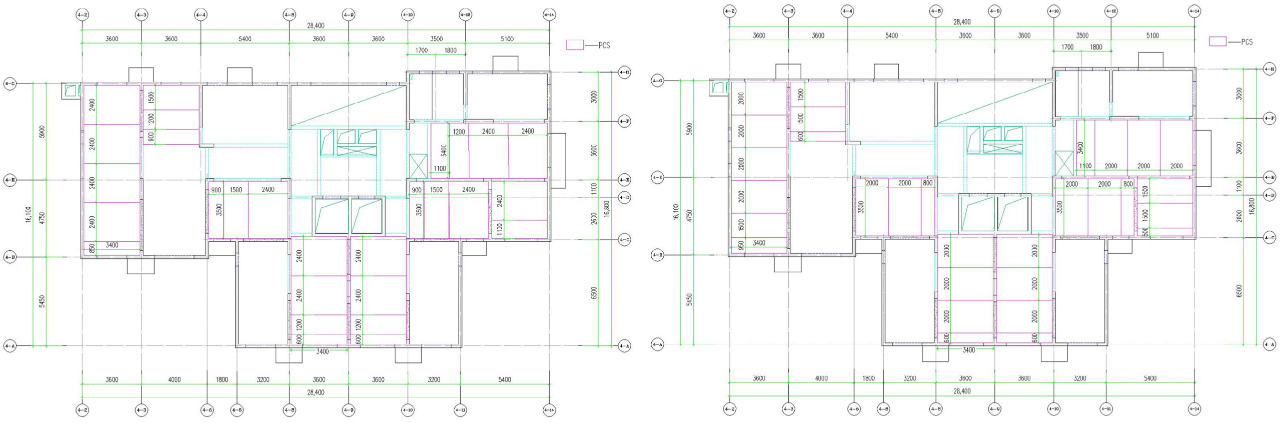
5.2. Comparison of Global and Collaborative Optimization Methods

In this subsection, the performance of the global optimization and that of the proposed multipopulation coevolution approaches are evaluated. Two methods are utilized to complete the PCS splitting design in eight subareas (subareas 1, 2, 5, 7, 8, 10, 11, and 12 in Figure 10), and the optimization success rate and obtained solutions were primarily compared. In order to further strengthen the reliability of the experiment, the parameter settings of the PSO algorithm needed to be unified. When the particle velocity v is large, the particle flight speed is fast and the exploration ability is strong, but the particle easily escapes from the optimal solution. When v is small, the flight speed is slow and the exploitation ability is strong, but the convergence speed is slow, and it is easy to fall into a local optimal solution. In order to balance the exploration and exploitation ability of the algorithm, it is essential to establish a reasonable speed range of the particle velocity. Meanwhile, when solving practical optimization problems, it is often desired to use a global search first to make the search space converge to a certain region, and then use a local fine search to obtain a high-precision solution. In this regard, based on the original recommended parameters, the adaptive correction method of the PSO parameters was employed. In the PSO algorithm, the inertia weight ω was linearly reduced from 0.9 to 0.4; learning rates lr_1 and lr_2 were set to 1.49; and the speed was limited to the interval $[-2, 2]$. Individual positions, based on integer coding, were restricted to the discrete integer solution space $\mathrm{Int}[0,\ 6]$. The population size P was set to 50, and the maximum number of iterations G was set to 1000. All the parameters' information is listed in Table 6, and these parameter settings were used for subsequent experiments.

Compared with global optimization, the optimization success rate of a collaborative optimization strategy based on decomposed subpopulations is higher in high-dimensional optimization problems. At the same time, the introduction of a collaborative optimization model can consider part of the global information and obtain better solutions. Figure 11 shows the splitting design of eight subareas using two methods. The optimization results using two methods are given in Table 7. Each method applying the splitting design was run independently 20 times to examine the success rate of the algorithm.

Table 6. Parameter settings.

Parameter	Values	Definition
lr_1	1.49	Learning rate for self-cognition
lr_2	1.49	Learning rate for group-cognition
ω	[0.4, 0.9]	Inertia coefficient
v	$[-2, 2]$	Velocity of the individual
x	Int [0, 6]	Location of the individual
P	50	Population size
G	1000	Maximum number of iterations



(**a**) (**b**)

Figure 11. PCS splitting results (unit: mm): (**a**) global optimization method; (**b**) multipopulation coevolution method.

Table 7. Comparative results of global optimization and collaborative optimization (20 runs).

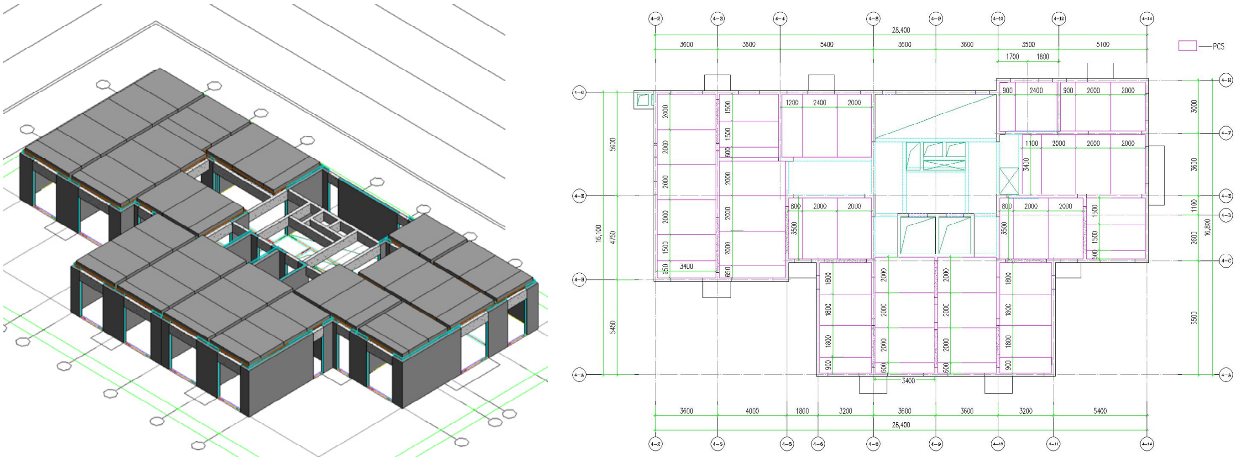
Parameter	Global Optimization	Collaborative Optimization
Number of successes	13	20
Optimal solution	[3, 15]	[2, 17]
Suboptimal solution	[3, 16]	[3, 15]

As shown in Table 7, the global optimization method achieved successful solutions in only 13 out of 20 independent experimental runs, while the collaborative optimization method ensured a successful resolution in all runs. Moreover, Table 7 presents the optimal and suboptimal solutions under successful outcomes for both methods. The solutions represent the types of board width spans and the total number of boards derived from solving the PCS splitting design problem. The first dimension of the solution represents the quantity of PCS specifications, while the second dimension represents the total number of PCSs. In the optimization process, individuals with fewer specifications and quantities of PCSs were selected as optimal solutions based on the design requirements. The results indicate that the global optimization method had difficulty guaranteeing to find feasible solutions and only one feasible solution was obtained, which was not necessarily the optimal feasible solution. In contrast to the global optimization approach, the multipopulation coevolution approach was more effective at finding the optimal sequence of solutions.

5.3. PCS Splitting Design for a Standard Floor

The splitting design of a standard floor was considered to assess the practicability and efficiency of the developed framework. The multipopulation coevolution PSO algorithm was implemented in the PCS splitting module based on the collaborative optimization model to provide the size, ownership, and placement order of each PCS. The parameter settings for the PSO were the same as those presented in Section 5.2. To further assess the applicability of the proposed collaborative optimization methods, comparisons were made with two other methods. The multipopulation noncollaborative method was used for the splitting design of the standard floor. Moreover, a parametric splitting design was also used for the splitting design of the standard floor and the width of all PCSs was set as 1200 mm.

The standard floor shown in Figure 10 was taken as the design object, and various methods, including multipopulation collaborative optimization, noncollaborative optimization, and parameter-based methods, were employed for the splitting design of PCSs. Before the experiments, the span grade of each rectangular region was defined. According to the PCS splitting design of the standard floor using the three methods, the IFC files and PCS layout drawings were automatically generated. With the aid of Autodesk Revit, BIM models including all PCSs in the standard floor are shown in Figures 12a–14a. As shown in Figures 12b–14b, the PCS layout drawings were automatically exported.



(**a**) (**b**)

Figure 12. Splitting design result of PCSs using the collaborative optimization method (unit: mm): (**a**) BIM models; (**b**) layout drawings.

The evaluation of the solutions was based on the quantity of PCS specifications and the total number of arranged PCSs recorded in the optimization process, serving as comparative indicators for each method. Table 8 presents the record design results for the three methods which indicate that each method had its characteristics. Considering the types of PCS widths, the parameter-based design method utilized a standardized PCS width, resulting in relatively fewer types. Considering the quantity of PCSs, the noncollaborative optimization method treated the optimal solution for each subarea as a local solution for the floor space, leaning towards a solution with fewer types of PCS widths. Considering the combination of specification and quantities of PCSs, a set of solutions with a balance of few PCS width types and a smaller PCS quantity was obtained. In the collaborative optimization method, the collaborative optimization strategy could comprehensively consider the consistent span types of split-board information and select solutions based on design requirements with fewer types or a smaller quantity of split boards. In the collaborative optimization method, considering the consistent span grades of subareas, solutions with fewer PCS width types and a smaller PCS quantity were generated.

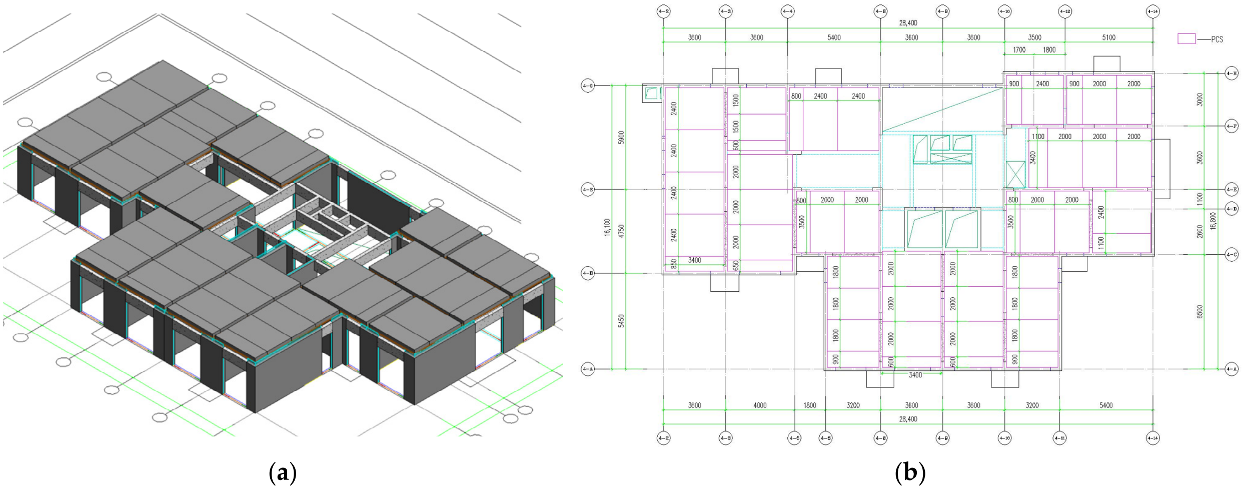


Figure 13. Splitting design result of PCSs using the noncollaborative optimization method (unit: mm): (**a**) BIM models; (**b**) layout drawings.

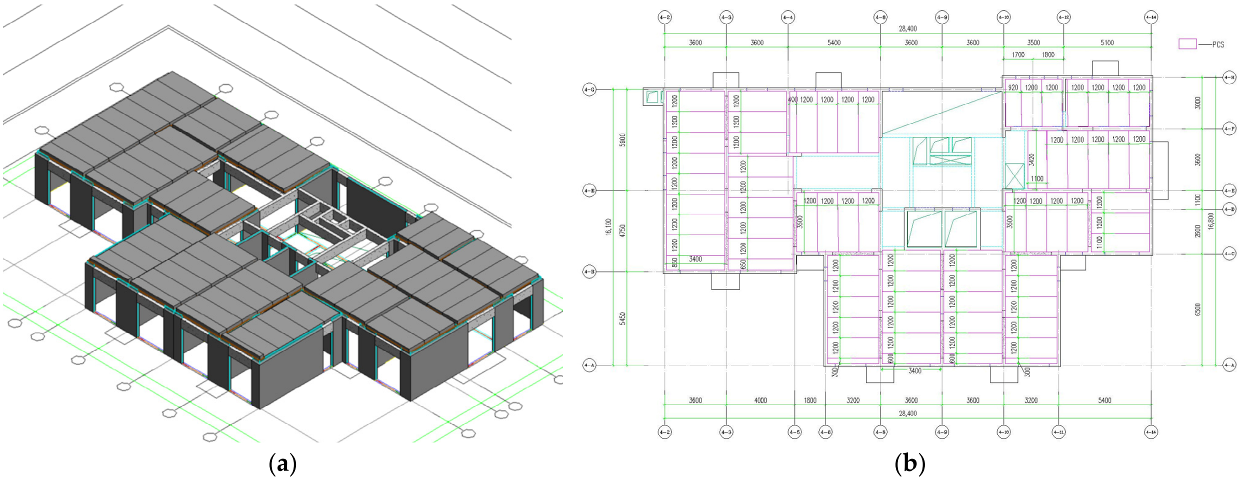


Figure 14. Splitting design result of PCSs using the parameter-based method (unit: mm): (**a**) BIM models; (**b**) layout drawings.

Table 8. Optimization results using different methods.

Span Grade	Collaborative Optimization Method	Noncollaborative Optimization Method	Parameter-Based Method
SL36	[2, 25]	[3, 23]	[1, 41]
SL33	[1, 6]	[1, 6]	[1, 10]
SL30	[2, 7]	[2, 7]	[1, 13]
DL39	[2, 5]	[2, 5]	[1, 9]
Total	[7, 43]	[8, 41]	[4, 73]

As shown in Table 8, there were three optimization results using three different design methods, namely, the collaborative optimization method, noncollaborative optimization method, and parameter-based method. The collaborative and noncollaborative optimization methods had consistent optimization effects in most subareas. However, in subareas with a higher number of other associated subareas, the collaborative optimization method tended to yield solutions with fewer specifications of PCS widths. Compared to the noncollaborative optimization method, the collaborative optimization method had fewer types of PCS widths, even though there were two additional PCSs. However, in the manufacturing process, the increase in types of PCS widths has a greater impact on production costs than having two more PCSs. Compared to the parameter-based design method, the collaborative optimization method had no advantage in terms of having fewer types of PCS widths. However, the parameter-based method also had the significant drawback of an excessive quantity of PCSs (73 PCSs in the table), leading to a sharp increase in transportation costs and labor costs during the production and assembly process. Therefore, the multipopulation collaborative optimization methods integrated the characteristics of the other two methods and generated more optimal solution sequences through balancing PCS specifications and quantities.

For the standard floor in a prefabricated building, five experienced detailing engineers were invited to independently complete the splitting design of PCSs. The average time required by the engineers was about 30 min to provide a PCS layout drawing and BIM model, whereas the total runtime of optimizing the splitting design and generating a drawing and BIM model completed using the proposed framework was about 30 s, which was significantly shorter. It is noteworthy that the PCS splitting design generated by an engineer cannot guarantee an optimal cost. Therefore, the proposed framework offers a more efficient design.

6. Conclusions

In this study, an intelligent BIM-based framework for the splitting design of PCSs was proposed to provide the PCS layouts in a building and obtain the PCS take-offs. The proposed framework considered the PCS layouts in the whole building to minimize the specifications and quantities of PCSs, reducing the costs in manufacturing, transition, and installation processes. Integrating the necessary data required from the BIM model or plan drawing, a PCS splitting module was developed to determine the sizes of all PCSs. A collaborative optimization model using a multipopulation coevolution algorithm was proposed to efficiently find the optimal splitting design of PCSs. According to the optimization results, the BIM models complying with the IFC standard were generated automatically, and the PCS layout drawing was exported to provide the size and position of each PCS.

To verify the proposed collaborative optimization model, a multipopulation coevolution PSO algorithm was considered in the automatic PCS splitting optimization. The splitting design of a standard floor in a prefabricated building was realized to validate the practicability and efficiency of the proposed framework. Based on this study, the following conclusions can be drawn:

- The collaborative optimization approach was implemented by dividing the original population into multiple subpopulations, which improved the accuracy and efficiency of PCS splitting.
- Compared with the global optimization method, the proposed multipopulation co-optimization strategy can effectively reduce the problem's dimensions and achieve a co-optimization among subpopulations, and a set of better solutions are efficiently solved.
- Compared with the parameter-based method and noncollaborative optimization method, the proposed method can solve the combined solution with better cost performance. This is because it takes into account the information on the type of board width and the number of split boards in the collaborative optimization process.

However, the developed intelligent splitting design method for PCSs has the following limitations:

- Serval specifications of standard PCS were assumed in this study, which limited the scale of the solution space.
- In this study, the proposed splitting design method was only applied for PCSs. For other prefabricated components such as walls, the optimization model should be further improved and extended in future work.

- In future work, more specifications of PCSs and joints will be added, and the range of values for the variable will be expanded to enhance the practicality of the PCS splitting method. Additionally, the splitting design approach of other prefabricated components, such as beams and walls, needs to be further studied.

Author Contributions: C.X.: conceptualization, writing—original draft, writing—review and editing, funding; X.Z.: methodology, validation, writing—review and editing; Z.W.: conceptualization, writing—review and editing; C.Z.: methodology, validation, writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by National Natural Science Foundation of China (project no. 52208219, 52178271) and National Key R&D Program of China (project no. 2021YFF0500903), to which the authors are very grateful.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available. Tables and figures represent data processed in a case study.

Acknowledgments: The authors would like to thank the referees for their constructive comments that improved the presentation as well as the content of the paper.

Conflicts of Interest: We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, and there are no professional or other personal interest of any nature or kind in any product, service, and/or company that could be construed as influencing the position presented in, or the review of the manuscript entitled.

Glossary

Key terms	Definition
PCS	Precast concrete slab.
W12	The nominal width of the unidirectional or bidirectional slab was 1200 mm (*W15*, *W18*, *W20,* and *W24* had the same definition as *W12*).
SL27	The nominal span of the unidirectional slab was 2700 mm (*SL30*, *SL33*, *SL36*, *SL39*, and *SL42* had the same definition as *SL27*).
DL30	The nominal span of the bidirectional slab was 3000 mm (*DL30*, *DL33*, *DL36*, *DL39*, and *DL42*, *DL45*, *DL48*, *DL51*, *DL54*, *DL57*, and *DL60* had the same definition as *DL30*).
R^{k}_{total}	Specification quantities of PCS.
g_{sum}	Number of slab span grades.
R^{g}_{k}	Span grade of the gth PCS.
w_{g}	Weight coefficient of the span grade width types of the gth slab.
R^{n}_{total}	Quantities of PCSs.
f_{sum}	Quantities of separating subareas.
R^{f}_{n}	Number of PCSs arranged in fth subareas.
$f^{l}_{best}(x_i)$	Optimal solution in subarea i.
$f^{subi}_{best}(x_{subi})$	Combined optimal solution for the ith subpopulation.
E^{k}_{total}	Width specification of the nonstandard PCS.
$f^{l\_list}_{best}\left(x_{subi\_j}\right)$	Selected optimal solutions in subarea j.
$f^{g}_{best}(X)$	Global optimization objective.
$f^{subi\_list}_{best}(x_{subi})$	Optimal solution of the ith subpopulation.
single-directional slab	Slab where the ratio of the long side to the short side is greater than or equal to 3.
bi-directional slab	Slab where the ratio of the long side to the short side is less than 3.
AEC	Architecture, engineering, and construction.
BIM	Building information modeling.
DfMA	Design for manufacturing and assembly.
DXF	Drawing Exchange Format.
MOO	Multiobjective optimization.
PSO	Particle swarm optimization
IFC	Industry Foundation Classes.

References

1. Xu, A.; Zhu, Y.; Wang, Z.; Zhao, Y. Carbon emission calculation of prefabricated concrete composite slabs during the production and construction stages. *J. Build. Eng.* **2023**, *80*, 107936. [CrossRef]
2. Li, Q.; Yang, Y.; Yao, G.; Wei, F.; Xue, G.; Qin, H. Multi-object Real-Time Automatic Detection Method for Production Quality Control of Prefabricated Laminated Slabs. *J. Constr. Eng. Manag.* **2024**, *150*, 05023017. [CrossRef]
3. Xin, G.; Long, G.; Zhao, J.; Zhang, Z. The Effect of Steel Trusses on the Mechanical Performance of Laminated Precast Slabs. *Buildings* **2023**, *13*, 1653. [CrossRef]
4. Tam, V.W.; Tam, C.M.; Zeng, S.X.; Ng, W.C. Towards adoption of prefabrication in construction. *Build. Environ.* **2007**, *42*, 3642–3654. [CrossRef]
5. Li, Z.; Li, T.; Wang, C.; He, X.; Xiao, Y. Experimental study of an unsymmetrical prefabricated hybrid steel-bamboo roof truss. *Eng. Struct.* **2019**, *201*, 109781. [CrossRef]
6. Díaz, G.; Herrera, R.F.; Rivera, M.L.; Atencio, E. Applications of Generative Design in Structural Engineering. *Rev. Ing. Constr.* **2021**, *36*, 29–47. [CrossRef]
7. Globa, A.A.; Ulchitskiy, O.A.; Bulatova, E.K. The Effectiveness of Parametric Modelling and Design Ideation in Architectural Engineering. *Sci. Vis.* **2018**, *10*, 99–109.
8. Banihashemi, S.; Tabadkani, A.; Hosseini, M.R. Integration of parametric design into modular coordination: A construction waste reduction workflow. *Autom. Constr.* **2018**, *88*, 1–12. [CrossRef]
9. Zou, Y.Q.; Sun, Z.C.; Pan, H.; Tu, W.L.; Dong, D.D. Parametric Automated Design and Virtual Simulation of Building Machine Using BIM. *Buildings* **2023**, *13*, 3011. [CrossRef]
10. Wang, Y.W.; Yuan, Z.M.; Sun, C.S. Research on assembly sequence planning and optimization of precast concrete buildings. *J. Civ. Eng. Manag.* **2018**, *24*, 106–115. [CrossRef]
11. Yuan, Z.M.; Sun, C.S.; Wang, Y.W. Design for Manufacture and Assembly-oriented parametric design of prefabricated buildings. *Autom. Constr.* **2018**, *88*, 13–22. [CrossRef]
12. Gao, S.; Jin, R.Y.; Lu, W.S. Design for Manufacture and assembly in construction: A review. *Build. Res. Inf.* **2020**, *48*, 538–550. [CrossRef]
13. Lu, W.S.; Tan, T.; Xu, J.Y.; Wang, J.; Chen, K.; Gao, S.; Xue, F. Design for manufacture and assembly in construction: The old and the new. *Archit. Eng. Des. Manag.* **2021**, *17*, 77–91. [CrossRef]
14. Bakhshi, S.; Chenaghlou, M.R.; Rahimian, F.P.; Edwards, D.J.; Dawood, N. Integrated BIM and DfMA parametric and algo-rithmic design based collaboration for supporting client engagement within offsite construction. *Autom. Constr.* **2022**, *133*, 104015. [CrossRef]
15. Feng, L.X.; Peng, J.; Yang, S.G.; Xu, J. Research and development of prefabricated detailed design software based on intelligent construction. In Proceedings of the 2022 IEEE 21st International Conference on Cognitive Informatics & Cognitive Computing, Toronto, ON, Canada, 8–10 December 2022; pp. 143–147.
16. Dong, D.D.; Zou, Y.Q.; Pan, H.; Zhou, G.W.; Feng, Y.; Tang, Y.C. DFMA-oriented modular and parametric design and secondary splitting of vertical PC components. *Sci. Rep.* **2023**, *13*, 3457. [CrossRef]
17. Xiao, Y.W.; Bhola, J. Design and optimization of prefabricated building system based on BIM technology. *Int. J. Syst. Assur. Eng. Manag.* **2022**, *13*, 111–120. [CrossRef]
18. Karami, F.; Dariane, A.B. A review and evaluation of multi and many-objective optimization: Methods and algorithms. *Glob. J. Ecol.* **2022**, *7*, 104–119.
19. Tang, J.; Liu, G.; Pan, Q. A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 1627–1643. [CrossRef]
20. Zhang, Y.; Zuo, T.; Zhu, M.; Huang, C.; Li, J.; Xu, Z. Research on multi-train energy saving optimization based on cooperative multi-objective particle swarm optimization algorithm. *Int. J. Energy Res.* **2021**, *45*, 2644–2667. [CrossRef]
21. Lu, Y.; Li, B.; Liu, S.; Zhou, A. A Population Cooperation based Particle Swarm Optimization algorithm for large-scale multi-objective optimization. *Swarm Evol. Comput.* **2023**, *83*, 101377. [CrossRef]
22. Xu, B.; Gong, D.; Zhang, Y.; Yang, S.; Wang, L.; Fan, Z.; Zhang, Y. Cooperative co-evolutionary algorithm for multi-objective optimization problems with changing decision variables. *Inf. Sci.* **2022**, *607*, 278–296. [CrossRef]
23. Wang, B.; Ren, L.; Prado, J.D.; He, W.; Jin, H.; Jiang, Q.; Wang, X. An adaptive mechanism with cooperative coevolution and covariance for differential evolution. *IEEE Access* **2021**, *9*, 99890–99904. [CrossRef]
24. Ma, X.; Liu, F.; Qi, Y.; Wang, X.; Li, L.; Jiao, L.; Yin, M.; Gong, M. A multi-objective evolutionary algorithm based on decision variable analyses for multi-objective optimization problems with large-scale variables. *IEEE Trans. Evol. Comput.* **2015**, *20*, 275–298. [CrossRef]
25. Du, A.; Chen, Y.; Zhang, D.; Han, Y. Multi-Objective Energy Management Strategy Based on PSO Optimization for Power-Split Hybrid Electric Vehicles. *Energies* **2021**, *14*, 2438. [CrossRef]
26. Fan, X.; Gao, Z.; Ji, W.; Guo, L.; Lin, W.; Wang, J. Thermodynamic optimization with multi objectives and parameters for liquid air energy storage system based on the particle swarm optimization (PSO). *J. Energy Storage* **2021**, *41*, 102878. [CrossRef]
27. Jain, M.; Saihjpal, V.; Singh, N.; Singh, S.B. An Overview of Variants and Advancements of PSO Algorithm. *Appl. Sci.* **2022**, *12*, 8392. [CrossRef]

28. *GB50010-2010*; Code for Design of Concrete Structures. Chinese Standard. China Architecture and Building Press: Beijing, China, 2010. Available online: https://www.chinesestandard.net/PDF/BOOK.aspx/GB50010-2010 (accessed on 23 January 2024).
29. *15G366-1*; Truss Reinforced Concrete Composite Slab. Chinese Standard. China Architecture & Building Press: Beijing, China, 2015. Available online: https://www.chinesestandard.net/PDF/BOOK.aspx/15G366-1 (accessed on 23 January 2024).
30. Srikanth, P.; Rao, C.S.P.; Janardhan, G.R. Algorithms for design feature data extraction from DXF files. *IEEE Trans. Evol. Comput.* **2009**, *4*, 1185–1196.
31. Lin, Y.H.; Liu, Y.S.; Gao, G.; Han, X.G.; Lai, C.Y.; Gu, M. The IFC-based path planning for 3D indoor spaces. *Adv. Eng. Inform.* **2012**, *27*, 189–205. [CrossRef]
32. Wikipedia. Available online: https://en.wikipedia.org/wiki/AutoCAD (accessed on 23 January 2024).
33. Deng, W.; Zhao, H.; Zou, L.; Li, G.; Yang, X.; Wu, D. A novel collaborative optimization algorithm in solving complex optimization problems. *Soft Comput.* **2017**, *21*, 4387–4398. [CrossRef]
34. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the IEEE/ICNN'95 International conference on neural networks, Perth, WA, Australia, 27 November 1995; Volume 4, pp. 1942–1948.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.