International Journal of Innovative Technology and
Interdisciplinary Sciences
https://journals.tultech.eu/index.php/ijitis
ISSN: 2613-7305
Volume 9, Issue 2
DOI: https://doi.org/10.15157/IJITIS.9.2.1049-1068
Received: 10.04.2026; Revised: 22.05.2026; Accepted: 28.05.2026

Research Article

Modular Small/Medium Scale Architectural Shading Structures Based on Computational Design

Prodromos Minaoglou[1], **Anastasia D. Stavridou**[1], **Marek Dudek**[2], **Panagiotis Kyratsis**[1]*

[1]Department of Products & Systems Design Engineering, University of Western Macedonia, Kozani, Greece

[2]Faculty of Management, AGH University of Krakow, Krakow, Poland

***pkyratsis@uowm.gr**

Abstract

Traditional methods of designing small and medium architectural structures are significantly limited in the complexity and parameterization of the result. Computational design uses algorithms to automate the design steps under certain rules and parameters, thus significantly upgrading the design methods of architectural structures. This study presents an automatic design application for modular small/medium architecture shading structures. This system is a parametric mesh consisting of bars and connectors. The user defines an initial geometry based on which the architectural structure is automatically designed via the computational design principles. In addition, the user can fully parameterize the result requiring minimum time by changing the input parameters via a user-friendly environment. As a result, the application automatically generates all the 3D computer aided design (CAD) models, the 3D assembled models, assembly instructions for the technical personnel, the BOM (bill of material), photorealistic views, appropriate files for prototyping via 3D printing and all the appropriate technical drawings with the basic technical dimensions. The design application was tested on various input geometries without any problem during the automatic design procedure. Prototyping of the designed structures was performed by 3D printing. The prototype presented in this research was 3D printed at a scale of 1:20. Based on the assembly instructions automatically generated by the application, the assembly was successfully completed.

Keywords: Computational Design; Small/Medium Architecture; Architectural Structures; Digital Fabrication; Optimization

---

INTRODUCTION

In recent years, in modern urban areas, there has been a great need for the construction of outdoor shading structures. Existing structures present two main problems. The first problem is the design and construction process, which must be repeated to adapt the

https://doi.org/10.15157/IJITIS.9.2.1049-1068

design to the area or to remain the same without its uniform composition of the surrounding area. The second problem is the cost of construction and the inability to recycle the structure by reusing it in a new position [1-3]. Each location has different requirements that are influenced by factors such as orientation, temperature, size and the climate of the area. A lack of structural optimization is often observed, resulting in designs that demand an increased number of resources. The integration of parametric concepts in the design process of architectural structures aims to solve the problems mentioned above. Computational design is the method that combines programming with 3D CAD (computer-aided design) design tools [4-6]. In essence, computational design using algorithms can transform a static architectural design into a fully parametric system. The parametricity of the system is defined in terms of input data that can directly or indirectly affect the result. These are processes that can automate one or more series of design steps, minimizing the design implementation time. Another advantage of computational design is the complexity of the result. Algorithms are based on mathematical concepts, developing patterns and geometric shapes of high complexity that are impossible to design with traditional design tools [7-10].

The development and creation of complex and intricate structures is often referred to as generative design, encouraging designers to produce free-form shapes. At the same time, new computational systems can better approach the entire design process, promote understanding of a building lifecycle from the initial idea to its demolition and recycling [11]. Modern digital tools such as Rhino™, Grasshopper™ not only require computer-based knowledge and handling, but also the designer's integration into a design thinking framework. An integral part of modern research is the connection between digital and physical fabrication through technologies such as additive manufacturing [12]. Authors in [13] used visual programming for designing architectural structures based on flower shapes. The geometries and patterns of specific flowers were transformed into structured geometric entities. The process and tools for deconstructing complex shapes using computational design were mostly examined. The results demonstrated that the use of modern tools such as computational design, can produce unique aesthetic and functional solutions.

Stefańska analysed the process of designing and fabricating lattice housing using coated metal lattice structures. More specifically, the conversion of curved surfaces into triangular and square flat panels was evaluated with the aim of saving materials and making the structure easier to construct. The design process of the structures was carried out via computational design. The main problem that emerged from the analysis was that the curved surfaces were difficult to construct when the panels were not flat. At the same time, it was shown that increasing the accuracy of converting curved surfaces into flat panels, the reduction in materials and construction time increased as well [14]. Authors in [15] also optimized lattice structures by changing their shape using computational design tools. Applying an algorithm and mathematical parametric formulas, 15 different shapes for a small-scale shelter were developed. The physical forces producing the different formations were simulated using the Form-finding and Dynamic Relaxation methods. The main objective of the study was to find the shape that best distributes the loads with the least use of material and maintaining the architectural aesthetics. The conclusions of the study showed that the preliminary optimization of the digital structure through specialized algorithms significantly reduced the weight of the structure and its environmental footprint.

Authors in [16] presented the design and fabrication of a temporary, low-cost and ecological pavilion structure. The aim was to create a structure with materials that offer a low energy footprint. Through the Form-finding method and computational design tools, a set of shapes was developed and evaluated by the designers. The selected shape was fabricated from materials that would have a limited lifespan, while maintaining a low energy footprint. Authors in [17] presented the process of designing and fabricating wooden pavilions using computational design principles. The main purpose was to adapt the fabrication process to local construction machinery (fabrication-driven). The application could be used in areas with limited access to advanced fabrication technologies, such as CAD/CAM (computer-aided design and manufacturing). The study combined computational design tools with traditional carpentry in each area. A significant advantage of smart and optimized design is the unnecessary use of specialized fabrication machinery, resulting in a reduction of the overall costs.

Researchers in [18] developed full-scale research about pavilion design consisting of wood and polymer concrete. This was a hybrid fabrication technique consisting of composite joints in the interior of the structures. The composition of the parts and the method of placement were determined using parametric and computational design tools [18]. Authors in [19, 20] designed a canopy that can interact with the environment. The smart canopy can move the mirrors that compose it with mechanisms. In the study, the system was developed using computational design and programming that operates both digitally and physically. The materials used were recyclable, thus promoting the sustainability principles. The resulting shade alters, depending on the inclination of the mirrors that control the solar radiation in each different condition of the day through appropriate sensors.

Authors in [21] developed a digital application for managing folded architectural structures. The study utilized parametric tools and structural analysis software. The digital application can optimize the amount of material used based on the characteristics of the geometry in question. The study does not perform an extensive analysis of how the folds work but approaches the folded architectural structure in a simplified way. Due to the simplification, there is a large discrepancy between the digital and the real geometry. Based on the application, digital tests were carried out, proving that the proposed methodology works for general geometrical forms. Authors in [22] applied the capabilities of parametric design to the construction of corrugated architectural structures. The authors developed an application that could solve the complex structures of the joints, based on mathematical models. A prototype was successfully constructed on a 1:4 scale.

In summary, the current literature demonstrates that modular architectural design has evolved from a static method of designing and producing standardized, flexible spaces to a multidimensional technological field (parametric systems) [23, 24]. While ensuring the structural integrity and durability of the units remains a fundamental requirement, the modern State-of-the-Art focuses on the full digital integration of the processes [25]. Through parametric and computational design, Generative Design, and BIM (Building Information Modelling), rapid optimization and automation of modular systems in the early stages are now achievable, dramatically increasing construction efficiency [26-28]. At the forefront of this evolution is the connection of modular architecture with Artificial Intelligence and digital construction. The introduction of machine learning methods for co-intelligent design, combined with technologies such as hybrid 3D printing, enables the creation of holistic, immediately adaptable solutions that respond to complex challenges such as post-disaster housing [29, 30]. Consequently, modular architecture is no longer seen simply as a construction logic, but as a dynamic, intelligent, and fully configurable system.

Research Gap and Contribution

The review of the existing literature highlights three main research gaps. The first one is that parametric tools, although very useful for individual design processes, lack the ability to automate the entire process for the simultaneous development of BOMs, instructions, fabricated files, etc. The second research gap concerns the requirement that designers specialize in programming general purposes CAD systems. There is a great need for applications that automate and simplify processes so that the designer has an upgraded role. The third research gap is the lack of quantitative comparisons between different structures, as there are no automated design processes within a parametric framework. While previous research focuses only on the development and production of the form or on individual fabrication stages, the present work deals with a comprehensive interactive system that can produce ready-made fabrication files, emphasizing the minimum set of links and the user's own shading optimization.

The present paper deals with a modular small/medium scale architectural shading structure using computational design tools. The main contribution of this paper is a parametric computational framework with quantified efficiency gains, evaluated through practical proof-of-concept applications and comparative analysis against traditional workflows. The algorithm was developed via Grasshopper3D™ software, which is an extension of Rhino3D™. Initially, the user defines a geometry on which the design of a small/medium scale architectural structure would be based on. Using both visual and textual programming, an architectural structure of octahedron shapes is automatically developed. After various modifications by the user and the definition of the shading areas, all the 3D CAD models, 3D CAD assemblies, bill of materials (BOM), the assembly instructions of the architectural structure, 3D printing files for prototyping etc. are automatically generated. The users can parameterize many design and fabrication characteristics via the user interface introduced. In Table 1, a comparison is made between the proposed method and the most modern approaches.

Table 1. Comparison of the proposed method with state-of-the-art approaches.

Reference / Method	Features supported	Automation level	Evaluation depth	Metrics used
[15]	Form-finding algorithms, dynamic relaxation, parametric optimization	Medium: Use of generative algorithms and cross-section optimization	Two stages: Comparison of 15 different geometric variations	Total construction weight, Weight per square meter of surface
[16]	Tensile structures, ecological materials	Medium: Parametric design and static analysis	Prototype: Form-finding, construction and assembly	Tensile forces, deformations, total budget cost
[17]	Discrete timber structures, fabrication-based design (FBD), side-to-side joints	Medium: Parametric design and static analysis	Multiple cases: Evaluation through 3 different stands constructed	Number of elements, Total meters, Construction time, Number of participants
[19]	Interactive smart canopies, responsive mirrors	High: Full integration of Grasshopper with Arduino	1:1 prototype: Simulation and control of motor/sensor system behaviour	Light intensity, Servo motor movement angle, Voltage
Proposed Method	Modular structures, interaction, octahedron, exported files ready for construction.	High: End-to-end automation, from geometry to BOM and instructions generation.	Rigorous / Multiple cases: Evaluation across a range of different test geometries	Design time, Material volume, Shading coverage, Approximation error

Research Questions and Hypotheses

The primary objective of this research is to develop a fully automated, end-to-end parametric pipeline through computational design that converts generic 3D geometries into fabrication-ready mesh structures. The study focuses on automating each process. To evaluate this framework, the following research hypotheses were formulated:

- H1: The proposed computational framework successfully automates the translation of 3D geometries into discrete bar and node structures. At the same time, accurate bills of materials (BOMs) and complete assembly instructions are developed without operator intervention in the rest of the process.
- H2: The use of this design application reduces the overall design-to-fabrication time, performing the process almost instantaneously compared to traditional manual modelling workflows.

- H3: The application offers parameterization tools for controlling the design process, displaying in real time any modification of the process itself.

PROPOSED METHODOLOGY

The design application of this study consists of three main stages. The first stage concerns the management of input data. At the beginning, the application requires the definition of the basic geometry. The file format does not play a significant role, as the application just requires the use of a single geometry without holes. The size of the geometry does not affect the result, as the total size of the architectural structure will then be defined. In addition to defining the initial geometry in the design application, the user incorporates a series of parameters and settings related to the architectural structure. Next, a detailed report is created on the input data of the design application. The second stage of the design application concerns the management of input data. It is carried out through computational design processes, whose algorithms were implemented through programming languages such as Grasshopper3D™, Python™, and C#™. The code consists of subcodes (programming functions) that receive, process, and output data. The third stage deals with the data extracted from the design application. At the end of the process, the design of the architectural structure is completed, including all the details. The assembly instructions are automatically developed and saved through automatic processes, 3D CAD models and assemblies, the BOM (Bill of Materials) is exported, and photorealistic views of the architectural structure are produced using textures, materials, lighting, and environment. At the same time, the design application exports ready-to-print files in. STL (Stereolithography) format for direct fabrication with the 3D printing method, and all the appropriate technical drawing views with the main dimensions. Figure 1 depicts the three main stages of this study.

Herein, the development of the algorithm, as well as the main tools that contributed to the completion of the application, are mentioned. To start with, the design application consists of five basic algorithmic sections. The basic programming language of the application is a visual programming language, namely Grasshopper3D™. The development method is based on nodes that are connected to each other and form the code. In each algorithmic section, the addition and use of external programming languages such as Python™ and C#™, which are text-based programming languages, was required.

The five basic algorithmic sections are as follows:

a) Structure definition: In the first algorithmic section, the basic data in the design application are defined, such as the initial geometry. Based on that, the algorithm develops the architectural structure. The geometry is segmented according to a 3D repeating octahedron structure. The segmentation creates a structure of tubes and joint points, which overall approximates the shape and form of the imported geometry.

b) Connectors: In the second algorithmic part, the connectors are designed parametrically so that they are placed at each joint point of the initial structure. For ease of component development, only two types of connectors were designed. The two types of connectors can be used at each node (joint points) of the structure. The parametric design of the connector geometries allows for automatic change when using an alternative input geometry.

c) Design finalization: In the third algorithmic part, the design process of the architectural structure is completed. The design details are carried out with automatic computational design procedures. At this point, an early visual evaluation is carried out by the user. Through special parameters, the user can control the degree of approximation of the initial geometry so that the geometry defined at the beginning does not need editing.

d) Shade cover definition: In the fourth algorithmic section, a user interaction algorithm is implemented. The purpose of this algorithm is to integrate the user into the design process of the architectural structure. The interaction is carried out through the 3D Viewport and the cursor. The user can choose in which areas of the architectural structure a shade cover is needed (shade panel). The goal is to develop the visual effect from the integration of shade covers, as well as how they affect the shadow effect by simulating sunlight in the environment.

e) Instructions development: In the fifth algorithmic section, the 2D design process of the instructions is automated. The instructions concern the assembly stages of each architectural structure. The number of stages is defined by the height of the structure as it consists of levels that start from the bottom to the top.

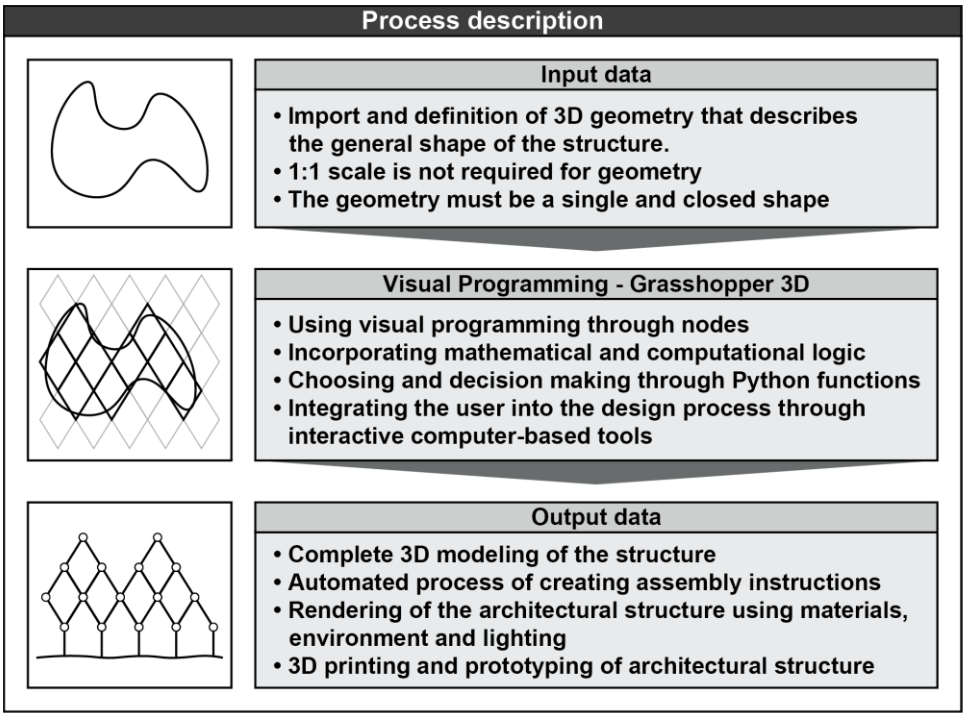


Figure 1. Description of the workflow based on its three main stages.

Figure 2 depicts the visual code as well as its algorithmic sections. To ensure readability and full clarification of the technical terms used in this article, Table 2 is provided. The summary table includes all necessary abbreviations, as well as mathematical symbols and variables used in the computational analysis and algorithm.

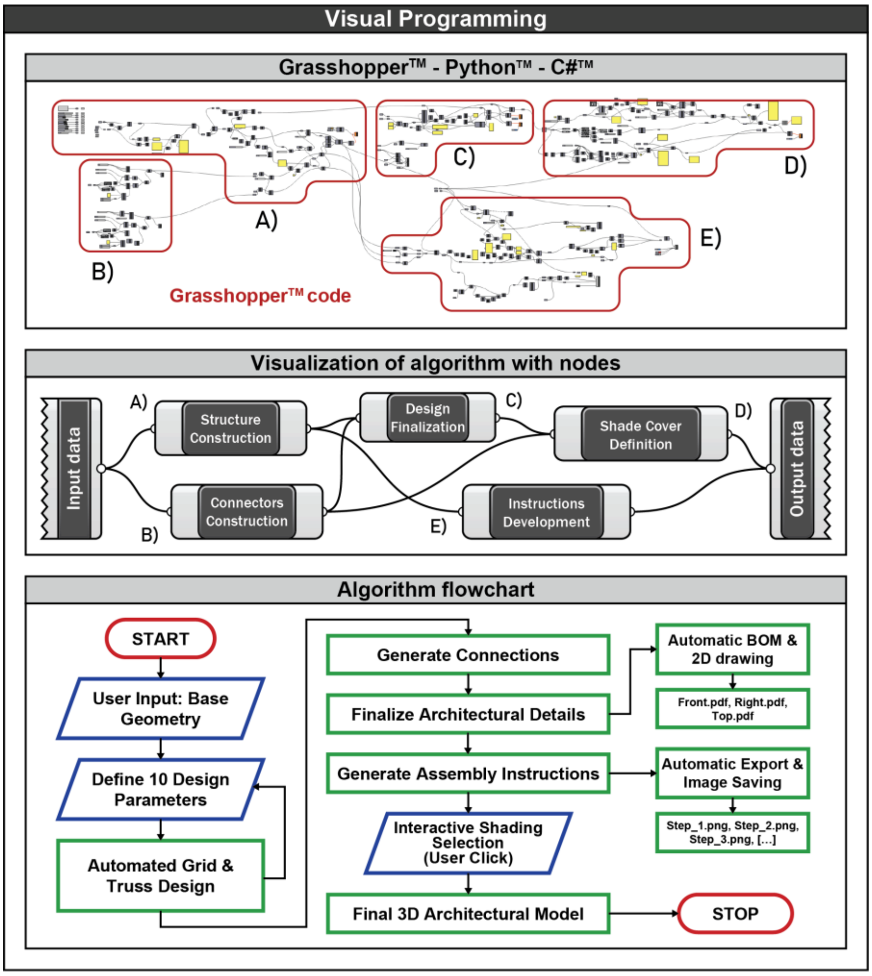


Figure 2. The code was developed in Grasshopper3D™.

Table 2. Nomenclature of abbreviations, variables, and mathematical symbols.

Abbreviation / Symbol	Description
BOM	Bill of Materials
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
STL	Stereolithography format
S	Final generated discrete structural system
G	Input 3D Boundary Geometry
P	User Parameter Set
L	Octahedron grid
η	Volume Fraction / Material Efficiency
$V_{structure}$	Total volume of the generated lattice structure
$V_{envelope}$	Total volume of the boundary geometry

Structure definition

The input data defines the basic properties of the design application. Based on the input data, the execution of sets of nodes is activated and deactivated. More specifically, it is an application control system in which the user can parameterize and fully control the results and operation of the design application. The application consists of 11 inputs. The first and most basic input of the application is the definition of the geometry on which the architectural structure will be based on. The definition process is achieved automatically through the appropriate node, which imports into Grasshopper3D™ the geometry loaded in the Rhino3D™ environment. Two equally important inputs are the total height of the structure and the diameter of the tubes. Based on these two parameters, the user decides the size needed for the architectural structure as well as the thickness of the available raw material. Based on these three inputs, the algorithm automates the process of designing the basic architectural structure. After the basic shape of the structure is finalized through the Edit Caps (On/Off) switch, the user can activate the process of defining the shading caps. By using the additional Reset Caps button, the user can delete the selection of the caps and start their definition from the beginning. At the same time, through the Light (On/Off) switch, a simulated light (sun type) is activated, where the algorithm calculates and displays the shadows created during the definition of the shading caps. The next switch, named People (On/Off), activates and deactivates the presence/appearance of 3D digital manikins in space. The main purpose is to better understand the size and shape of the architectural structure when compared with the dimensions of a human body. The last four inputs are related to the creation of assembly instructions. Initially, because the assembly process takes 3-5 seconds, they are disabled with the aim of running the application faster during the structure parameterization. The activation switch is called Instructions (On/Off), while the process of generating and storing the instructions is called Instructions (Start). Finally, the user can change the viewing angle of the assembly instructions 360 degrees around the architectural structure and the Zoom of the viewing angle. Figure 3 shows the 11 inputs of the design application as well as stages from the results of the algorithmic section A, such as the initial geometry, the initial resulting structure and the final shape of the structure. Also, Table 3 presents detailed information on each parameter.

Table 3. Basic system parameters and value ranges.

Variable name	Parameter category	Data type	Value range
Total Height (m)	Geometric	Numerical (Slider)	2.5m – 20m
Bar Diameter (mm)	Geometric	Numerical (Slider)	50mm – 90mm
Reset Caps	Processing	Logical (Button)	True, False
Edit Caps (On/Off)	Processing	Logical (Boolean)	True, False
Light (On/Off)	Display	Logical (Boolean)	True, False
People (On/Off)	Display	Logical (Boolean)	True, False
Instructions (On/Off)	Export	Logical (Boolean)	True, False
Instructions (Start)	Processing	Logical (Button)	True, False
Instructions Angle	Display	Numerical (Slider)	0° – 360°
Zoom	Display	Numerical (Slider)	100 – 700

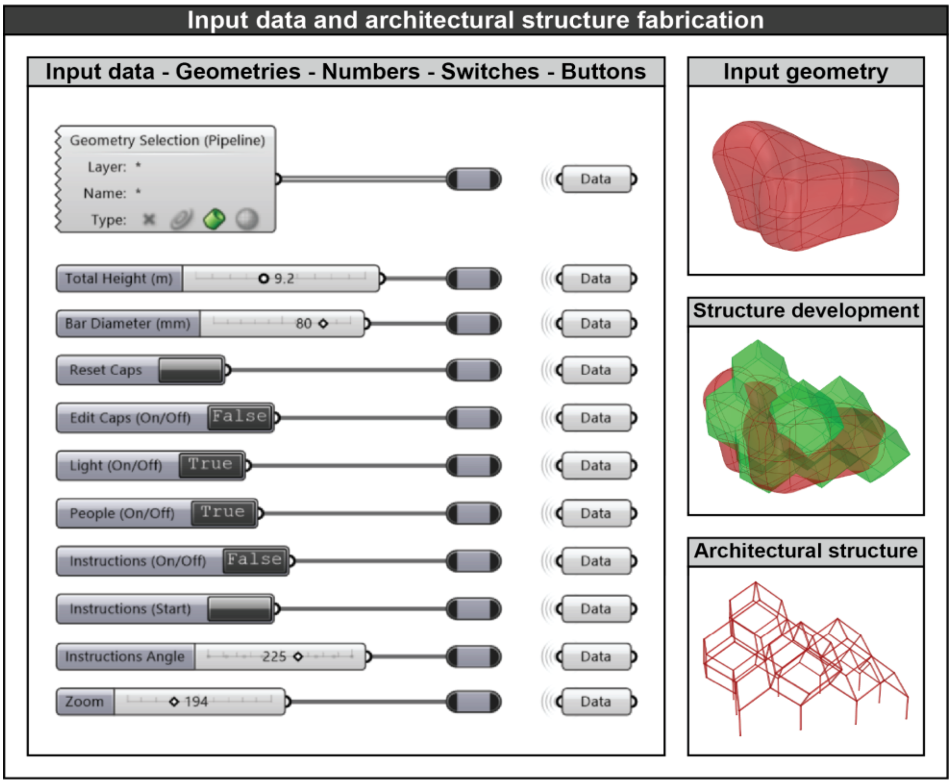
The conversion of the basic imported geometry is done through the following function (1):

$$S = f(G, P) \tag{1}$$

Where: G: The 3D geometry entered by the user, P: The vector of parameters defined by the user, and S: The final generated discrete structural system.

The process used by the algorithm is a two-level spatial delocalization. In the first level, the lattice generation is developed from octahedra L based on the user parameters P. With the function L = OctahedronGrid(P), the grid is created from the octahedra. In the second level, a Boolean intersection is performed between L and G, and the final structure S is generated (2).

$$S = L \cap G \tag{2}$$



Geometry Selection (Pipeline)
Layer: *
Name: *
Type:

Total Height (m) 9.2
Bar Diameter (mm) 80
Reset Caps
Edit Caps (On/Off) False
Light (On/Off) True
People (On/Off) True
Instructions (On/Off) False
Instructions (Start)
Instructions Angle 225
Zoom 194

Figure 3. The parameters that control the result of the code.

Connectors

The geometry of the connectors requires detailed specifications for proper placement and functionality. Based on the selected octahedron structure, the corners of each connector have the same inclinations (angles). The main purpose of the algorithm is the parametric design of the connectors. The input data of each connector is the total size and diameter of the holes and the retention thickness based on two parameters. In the context of the study, algorithms were developed for two different types of connectors, but the expansion of this option can be implemented. One type concerns the union of all connections with 8 holes. The other type deals with the connection of the structure to the base of the overall architectural structure using 5 holes. In this way, each architectural structure will maintain vertical columns at the bottom of it, based on which it will be supported on the ground. Figure 4 presents the four stages of the parametric design of the connector. In the first stage, the basic shape is defined using 5 and 8 lines. Based on the lines, the volume of the connector is created, and then the holes are made by subtracting geometries. Finally, the geometry is sliced peripherally to create flat surfaces around the connector. Also, Figure 4 presents the input data for the parametric design as well as four different variations based on different input data.

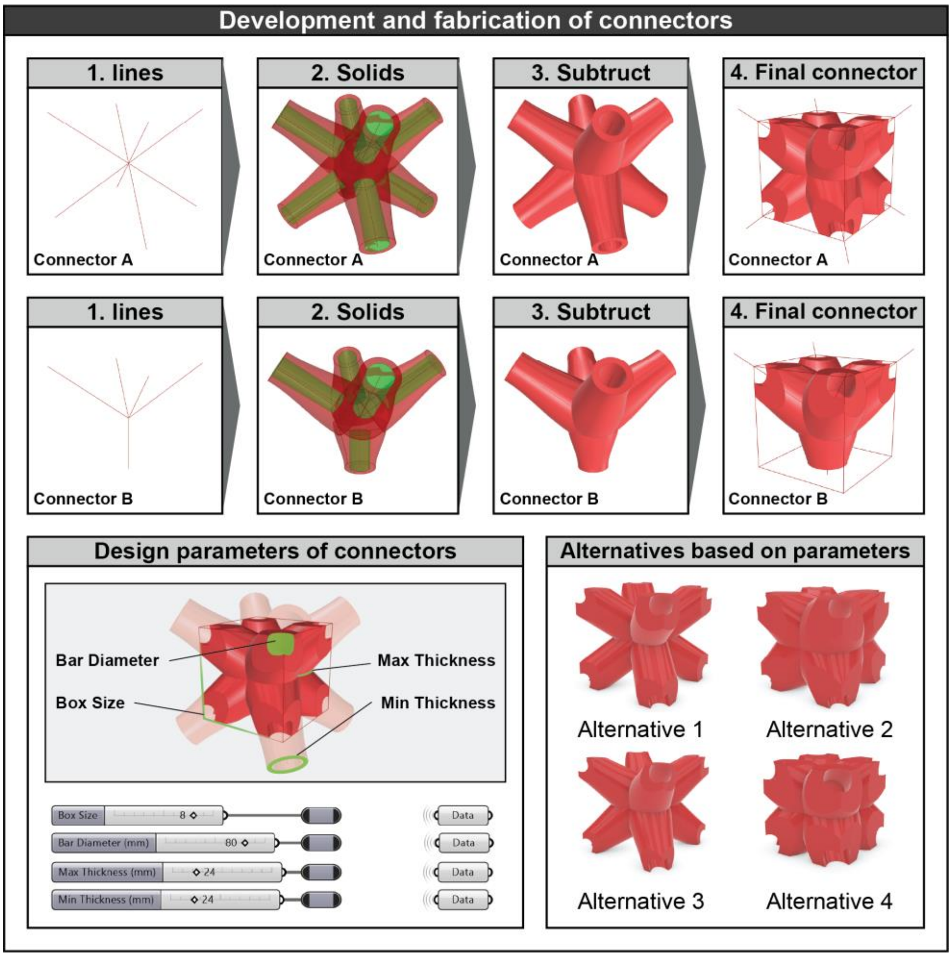


Figure 4. The two types of connectors used and their parameters.

Design Finalization

At this point, the digital assembly of all the components that make up the architectural structure takes place. The result consists of the tubes, the base tubes, the connectors, and the base connectors. At the same time, textures are defined on each component with different colors. By adding lighting and a specific viewing angle, the visual depth of the overall architectural structure is presented. The visual result helps in the early evaluation of the structure by both the designer and client. During the evaluation, the application operator can modify the initial geometry to change the result according to their preferences. After the assembly is completed, digital images of the architectural structure are automatically developed from various angles. In this way, details around the structure are presented. Three technical drawing plans are automatically produced, in which the overall dimensions of the architectural structure are presented. Figure 5 depicts the results of the assembly of the components of the architectural structure, the names of each component, and the technical drawing plans with the overall dimensions.

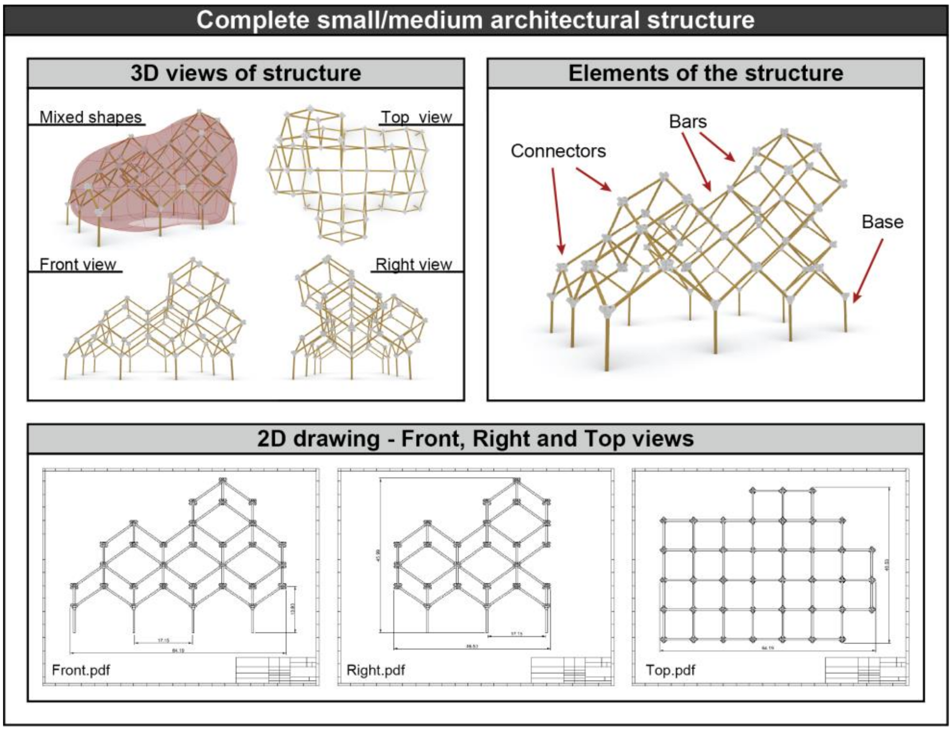


Figure 5. Completion of the architectural structure and development of 2D drawing views.

Shade Cover Definition

This application allows the user to select the areas wishing to place a shade cover. The definition is achieved through an interactive process in which the user, via the cursor, selects on the 3D architectural structure the empty areas needed covering. With a second click the area can be deselected. The shade cover was designed through a parametric algorithm so that it can be adapted to each empty area. In the present study, the shade cover of each area is constant, as it is a structure that repeats the same exact shape. The design details of the shade cover, such as the retention geometries, thickness and dimensions, are applied at the end of the empty area selection process. Design details are not implemented in real time due to build time limitations. The calculation time results from the number of shadings covers and can be from 10 seconds to 1 minute. This period is affected by the complexity of the structure and the computing power of the computer. The geometry of each shading cover allows the placement of many covers in a nearby area (adjacently connected) without overlapping the containment geometries. Figure 6 depicts the process of selecting empty areas. The selected areas are in blue, while the area over which the cursor is located for the next selection is shown in light blue. At the same time, some of the automated stages of designing a shading cover are presented (the method of placement next to an adjacent cover and the parameterization of the cover in different shapes of empty areas). Finally, after the selected areas are finalized, the result is displayed with the covers placed on the architectural structure with all the designed details. During the entire process, the user can see in real time the total shadow resulting from the architectural structure. The direction of the light starts from an initial location, which can be changed manually.

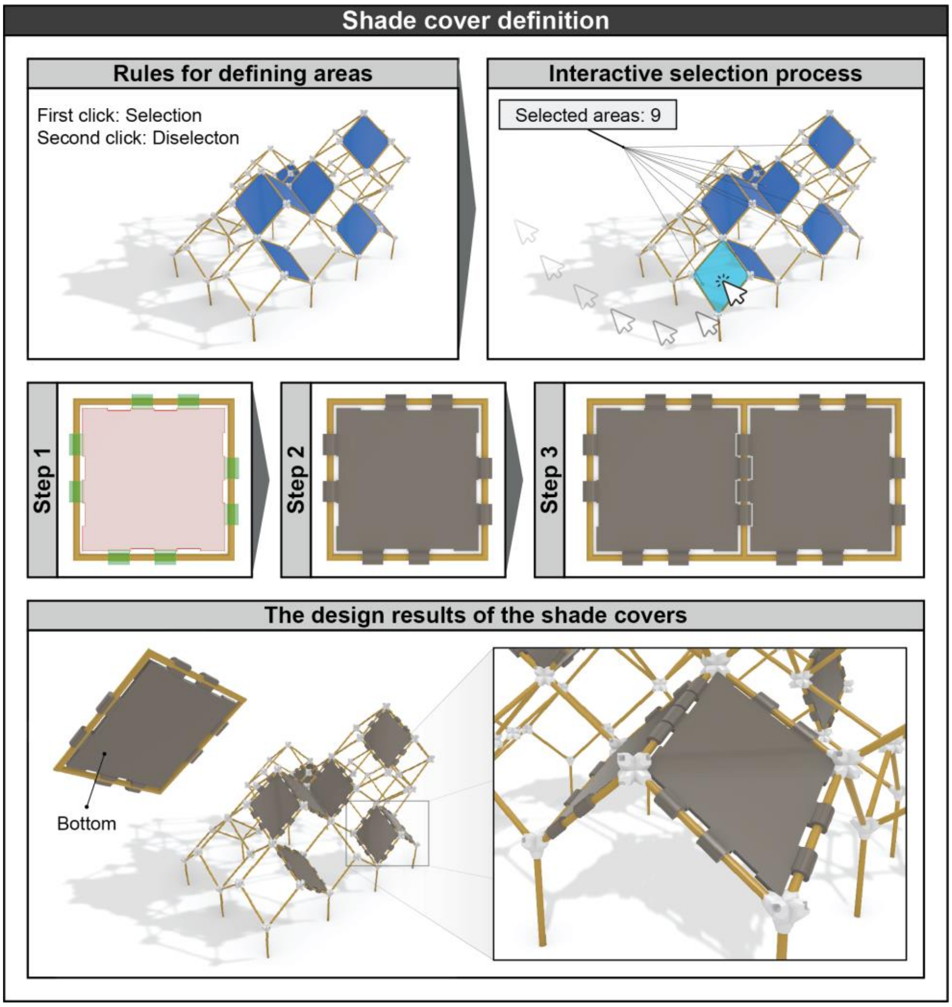
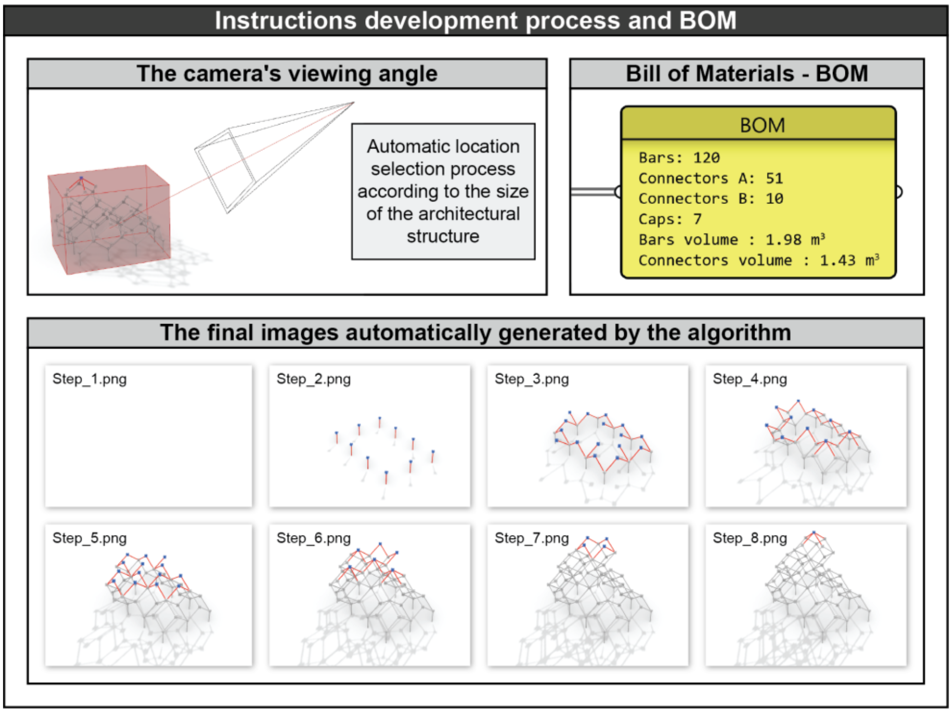


Figure 6. Interactive process of selecting areas to add shading covers.

Instructions Development

In the last section, the process of developing assembly instructions was automated. This process produces images from a specific perspective that facilitate the assembly of the architectural structure. Each image shows one of the assembly stages, starting from the first stage, which is the base. Each subsequent image uses the already assembled components in grey and adds the new components in the correct positions. The number of stages depends on the total height of the architectural structure. In the example presented in Figure 7, eight assembly stages were used. The perspective of each photo is automatically determined according to the dimensions of the architectural structure. The user can modify the zoom and the side from which the structure is viewed. All images are automatically saved in a folder defined. At the same time, the application offers the creation of a Bill of Materials (BOM) in which the components to be used and their quantity are presented. Finally, a calculation of the total volume of the architectural structure is performed, based on which the total weight and cost can be calculated. Figure 7 shows the camera angle, the BOM along with the total volume/weight, and the images from the assembly instructions stages.



```
Bars: 120
Connectors A: 51
Connectors B: 10
Caps: 7
Bars volume : 1.98 m³
Connectors volume : 1.43 m³
```

Figure 7. Automatic assembly instruction development process.

To quantitatively evaluate and document the stability of the proposed system, six analytical tests were conducted using different initial geometry morphologies (Figure 8). For each separate case (variant), the process is visualized through four design stages: from the input of the initial geometry and its conversion into an octahedral mesh, to the production of the final architectural structure. For each variant, three critical metrics were extracted: the volume ratio η (volume fraction between initial surface and final structure), the total computational execution time of the code, and the final produced components. In each structural variant, the result was produced without issues. The results confirm the adaptability of the tool to various architectural requirements.

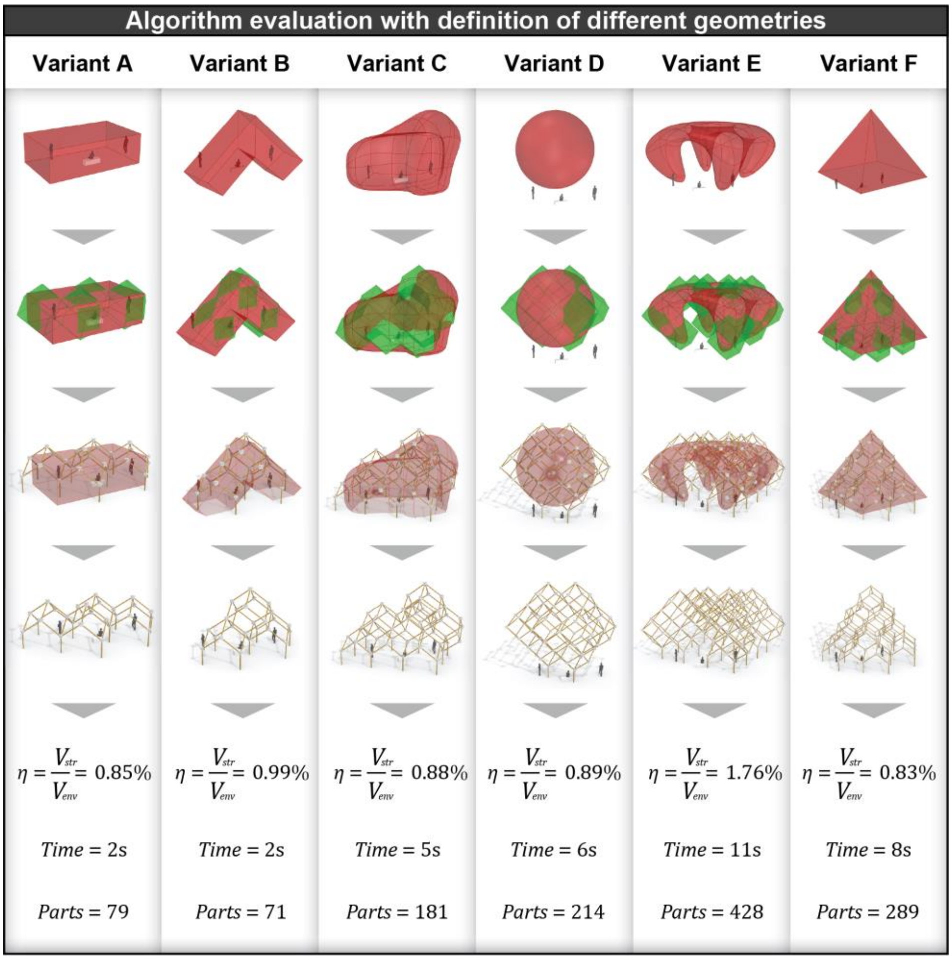


Figure 8. Visual and quantitative evaluation of six different inlet geometries.

Rendering and Prototyping

In the last stage of this study, the final visualizations are performed using advanced rendering tools. Materials, environment, and lighting are added to highlight the final architectural structure. In each illustration, human manikin shapes are positioned so that the size and proportions of the architectural structure become clear. Then, using additive manufacturing (3D printing), a prototype can be fabricated. Figure 9 shows images from the rendering of the architectural structure as well as images from the prototyping stages. With the prototype construction, the final evaluation of the result was carried out. More specifically, a better evaluation of the space as well as the way shadows appear in a real environment was carried out. Overall, the assembly process was easily carried out according to the generated instructions. The printer used in this study was the Creality™ K1 Max (Creality, Shenzhen, China), which has excellent accuracy and high speeds. A total of 61 connectors were printed (51 connectors and 10 base connectors) with a total printing time of 8 hours and 31 minutes. The material used was NEEMA3D™ PLA: EVO, a high-strength reinforced PLA. The prototype was printed at a scale of 1:20 with a weight of 165.7g and dimensions width = 460mm, height = 460mm, and depth = 630mm. The 120 bars were not manufactured using 3D printing but were made using ready-made Bamboo sticks of 4mm thickness and 100mm length.

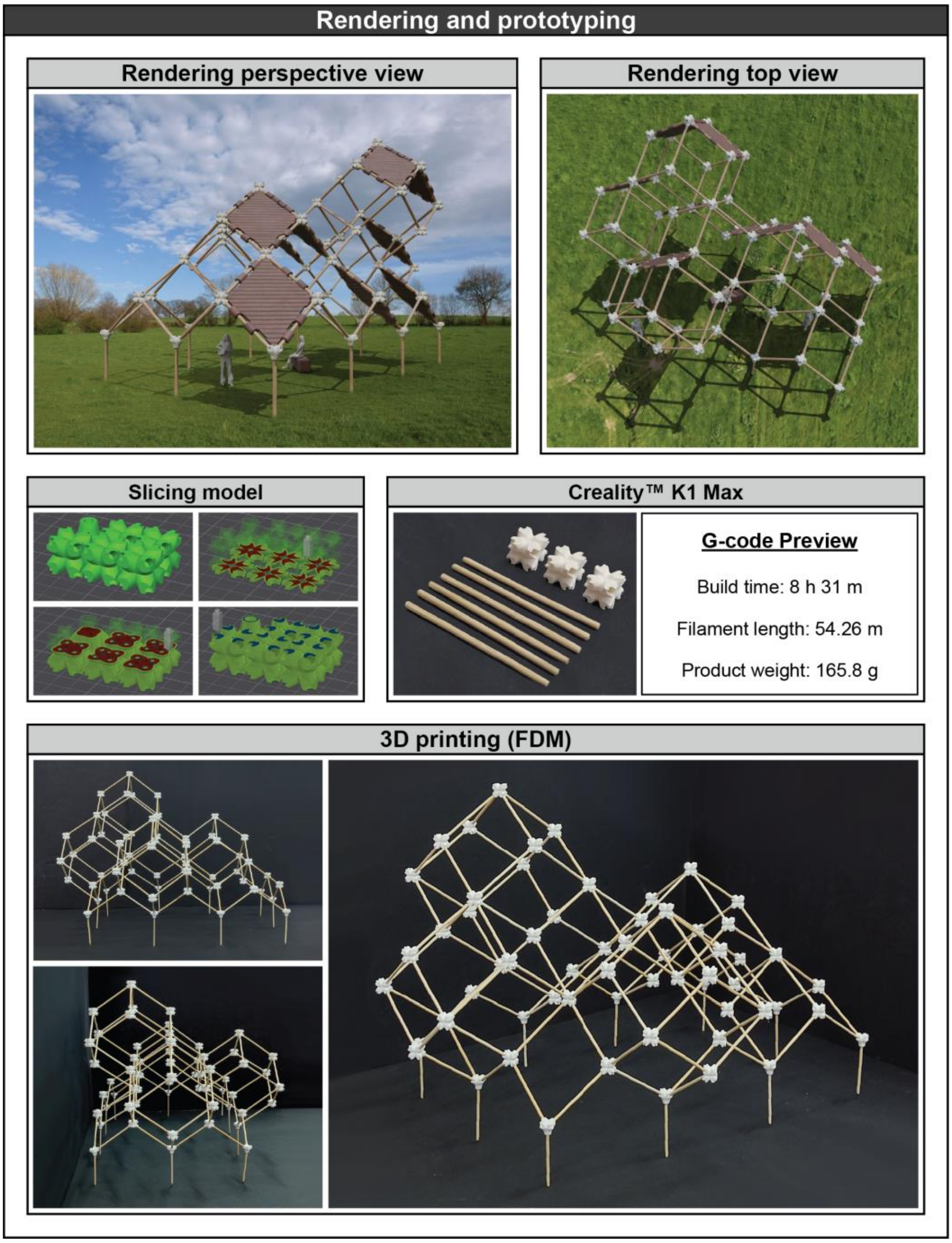


Figure 9. Rendering in a digital environment and 3D printing.

To quantitatively evaluate the proposed algorithm and confirm the research hypotheses of the study, key performance metrics were extracted. These data document the excellent production speed, optimized material efficiency (through volume fraction calculation), as well as the ease of final fabrication and assembly. The results are summarized in Table 4, demonstrating the practical applicability of the system.

Table 4. Quantitative performance metrics of the produced network.

Performance metric	Description	Unit / Type	Value
Volume fraction (η)	The ratio of the volume of the mesh to the volume of the original geometry.	$\eta = \frac{V_{structure}}{V_{envelope}}$	3.41 / 385.88 0.0088 (0.88%)
Design speed	The time required by the algorithm to fully generate the structure.	Seconds = (s)	3.8 s
Manufacturing time	Average 3D printing or production time per part (link).	Minutes (min)	22 min
Assembly complexity	Total number of distinct elements to connect (bars + nodes).	Count (Integer)	181 parts

DISCUSSION AND LIMITATIONS

The choice of octahedral shape proved to be particularly efficient to produce the modular structure, as it offers an optimal balance between spatial coverage. This shape can be repeated in space simultaneously in all 3 dimensions. This property of the shape makes it ideal for geometries that require repetition while maintaining the same lengths in the bars and the same angles between them. At the same time, the octahedral shape compared to the cubic shape (which can be repeated in space) is the development of geometries with greater complexity as its angles are not limited like the angles of the cubic shape. By extension, using the octahedral shape the architectural structure can much better approximate the initial imported shape.

Compared to recent bibliographic approaches analyzed in Table 1, the current literature focuses mainly either on the production of triangular meshes or purely on static optimization. In most cases, the construction of these structures requires a large number of different components for assembly. The proposed framework is significantly different as it uses only two types of connectors and one type of bar to create a structure very close to the imported shape. At the same time, a fully automated design process was presented as instead of a morphogenesis algorithm, the entire workflow is unified: from arbitrary geometry to the automatic extraction of a Bill of Materials (BOM) and assembly instructions. The innovation lies in "end-to-end" automation and interactivity, dramatically reducing the design time compared to traditional methods.

Regarding the limitations of the present study, the absence of an extensive statistical and quantitative comparison with alternative discretization methods is noted. Given that the primary goal of the article was the algorithmic foundation and the full automation of the process, conducting a rigorous experimental evaluation on a wider set of complex geometries, as well as the integration of structural analysis tools, are some of the next steps in future research.

SUMMARY AND CONCLUSION

In this study, a small/medium scale architectural structure design application was developed as a modular shading system. The design of this architectural structure was based on computational design tools. More specifically, the application automatically designs all components through the proposed algorithm. The application user initially defines a geometry based on which the architectural structure would be designed. At the same time, the user can modify the result of the architectural structure in real time. Through the algorithms, all design details are automated. One of the most important features of the design application is the automatic creation of the assembly instructions. In addition to the complexity of the geometry, the assembly steps are produced in the form of images in which the steps that must be followed are presented step by step. Upon completion of the application, the operator can define through an interactive process in which areas the shading covers should be placed. At the end of the process, the design application automatically generates the Bill of Materials (BOM), which contains the components to be used as well as their quantity. The 3D CAD models and assembly files are used for all the downstream applications i.e., rendering, prototyping.

The use of computational design and programming tools can significantly improve the design process of small/medium scale architectural structures. Through automatic processes, results are produced in a shorter period, reducing the possibility of human error. The role of the designer is transformed into a user role, who can influence the result through appropriate input parameters and make changes to the code if necessary.

The evaluation of the system fully confirmed the initial research hypotheses of the study as the full automation of the conversion of 3D geometries into bar/node lattices was achieved, with simultaneous automatic generation of Bills of Materials (BOM) and assembly instructions. At the same time, the time from design to production was reduced, as the process is performed almost instantaneously compared to traditional manual methods. Finally, parametric control of the process was ensured, with each modification being displayed immediately in real time.

Adaptability and extensibility are two characteristics that are significantly enhanced, when integrating computational design into the process. The integration of new extensions in future development of the design application can add additional features and control tools for the designer i.e., finite element analysis, static and dynamic loading calculations.

AUTHOR CONTRIBUTIONS

Conceptualization, P.M. and P.K.; methodology, P.M. and P.K.; software, P.M., A.D.S. and M.D.; validation, P.M., M.D. and P.K.; formal analysis, P.M. and A.D.S.; investigation, P.M., A.D.S. and P.K.; resources, M.D. and P.K.; data curation, P.M. and A.D.S.; writing—

original draft preparation, P.M. and M.D.; writing—review and editing, P.M., M.D., A.D.S. and P.K.; visualization, P.M. and M.D.; supervision, P.K.; project administration, P.K. All authors have read and agreed to the published version of the manuscript.

ACKNOWLEDGMENT

The authors would like to thank the Management Faculty of AGH University of Krakow for their financial support.

CONFLICT OF INTERESTS

` The authors confirm that there is no conflict of interest associated with this publication.

REFERENCES

1. Filz, G.H. Geodesic winding of minimal surfaces: the architecture, structure and geometry of an irregular grid-shell pavilion assembled from thin plywood strips. *In Proceedings of IASS annual symposia*, **2021**; pp. 1-12.
2. Groba, U., & Blumer, S. RE: source pavilion-exploring the circular use of wooden building materials. *In Proceedings of 13th world conference on timber engineering*, WCTE, Oslo, Norway, **2023**; pp. 3625-3633.
3. Seixas, M. A., Ripper, J. L. M., & Ghavami, K. Prefabricated bamboo structure and textile canvas pavilions. *Journal of the International Association for Shell and Spatial Structures*, **2016**, *57*(3), 179-188.
4. Caetano, I., Santos, L., & Leitão, A. Computational design in architecture: Defining parametric, generative, and algorithmic design. *Frontiers of Architectural Research*, **2020**, 9(2), 287-300.
5. Kyratsis, P., Aidinli, K., Minaoglou, P., & Firtikiadis, L. Computational interior design based on 2d patterns. *In International Symposium on Graphic Engineering and Design*, **2022;** pp. 21-27.
6. Manavis, A., Minaoglou, P., Tzetzis, D., Efkolidis, N., & Kyratsis, P. Computational design technologies for interior designers: a case study. *In IOP Conference Series: Materials Science and Engineering*, **2021**, *1009*(1), 012037.
7. Assasi, R. Parametric design, a historical and theoretical overview. *In Proceedings of the International Conference on Emerging Technologies in Architectural Design*, **2019**; pp. 1-7.
8. Eltaweel, A., & Yuehong, S.U. Parametric design and daylighting: A literature review. *Renewable and Sustainable Energy Reviews*, **2017**, 73, 1086-1103.
9. Al-Azzawi, T., & Al-Majidi, Z. Parametric architecture: the second international style. *In IOP Conference Series: Materials Science and Engineering*, **2021**, *1067*(1), 012019.
10. Farouk, A., Eldaly, H., & Dewidar, K. Parametric design as a tool for performative architecture. *Journal of Al-Azhar University Engineering Sector*, **2019**, *14*(50), 148-157.
11. Rokicki, W., & Stefańska, A. Architectural Ephemerids in Terms of Generative and Parametric Design. *International Journal of Structural and Civil Engineering Research*, **2019**, *8*(2), 146-151.
12. Manavis, A., Firtikiadis, L., Spahiu, T., Efkolidis, N., & Kyratsis, P. Parametric architectural design using shapes and structures. *Journal of Graphic Engineering and Design*, **2022**, *13*(4), 13-20.

13. Riyaz, M., & Anbu, M. Programación visual como herramienta para el cálculo de geometrías inspiradas en el florismo para el diseño de stands de exposición. *Revista Hábitat Sustentable*, 2025, *15*(2), 10–23 (Spanish Language).
14. Stefańska, A. Reticulated Roof Structures Optimisation Based of Triangular and Quadrilateral Planar Panels. *In Proceedings of the Creative Construction e-Conference*, **2020**; pp. 27-32.
15. Stefańska, A., Mikos, A., & Zientała, E. Geometrical transformations of reticulated pavilion canopies as a method for structural optimization. *In E3S Web of Conferences*, **2020**, *220*, 01085.
16. Latka, J. F., & Swieciak, M. The obverse/reverse pavilion: An example of a form-finding design of temporary, low-cost, and eco-friendly structure. *Buildings*, **2021**, *11*(6), 226.
17. Cáceres Corvalán, K., & Calvo Castillo, F. J. Crafting innovation: Discrete timber pavilions and fabrication-driven approaches. *International Journal of Architectural Computing*, **2024**, *22*(1), 89-107.
18. Gheorghe, A., & Vierlinger, R. DigDesFab15 research pavilion. *Frontiers in Digital Humanities*, **2017**, 4, 18.
19. Altameemi, A.S., & Jabbar, A. M. Using digital software to design interactive smart canopies for the outdoor environment. *Architecture and Engineering*, **2025**, *10*(2), 19-32.
20. Hoxha, A., & Qamili, A. Thermal Analysis of Polymeric Composite Panels for Sustainable Energy Applications. *International Journal of Innovative Technology and Interdisciplinary Sciences*, **2026**, *9*(1), 402-432.
21. Meyer, J., Duchanois, G., & Bignon, J.C. The fold, a tool of design architectural structures: Development of a structural and formal language in wood material. I*n Global Civil Engineering & Applied Science Conference*, **2014**; pp. 1-8.
22. Stavric, M., Wiltsche, A., & Bogensperger, T. Generative design for folded timber structures. *In Proceedings of the 20th International Conference on Computer-Aided Architectural Design Research in Asia*, **2015**; pp. 673-682.
23. Kim, J., Ryu, R., & Kim, Y. A basic study on unit module planning for the design diversity of modular architecture. *Asia-Pacific journal of convergent research interchange*, **2019,** *5*(1), 31-45.
24. Djukanovic, M., Alegre, A., & Teixeira Bastos, F. *Prefabricated solutions for housing: modular architecture and flexible living spaces. Buildings,* **2025**, *15*(6), *862*.
25. Lacey, A. W., Chen, W., Hao, H., & Bi, K. Structural response of modular buildings–an overview. *Journal of building engineering*, **2018**, *16*, 45-56.
26. Curletto, G. Parametric modeling in form finding and application to the design of modular canopies. *WIT Transactions on the Built Environment*, **2014**; *136*, 223-234.
27. Zárate, W., Figueiredo, B., Brandão, F. J. S., Pires, M., & Carneiro, P. Generative design of modular/industrial architectural system. In *5º Congresso Português de Building Information Modelling Volume 1: ptBIM*, **2024**; pp. 299-310.
28. Greenough, T., Smith, M., & Mariash, A. Integrating computational design to improve the design workflow of modular construction. *Modular and Offsite Construction (MOC) Summit Proceedings*, **2019**; pp. 165-172.
29. Sun, M., Buš, P., Chen, J., & Sun, K. Co-Intelligent Design Toward Modular Structures. In *ACCELERATED DESIGN-Proceedings of the 29th CAADRIA Conference, Singapore*, **2024**; pp. 171-180.
30. Mehdizade, A., & Ayoobi, A. W. A Holistic Design Framework for Post-Disaster Housing Using Interlinked Modules for Diverse Architectural Applications. *Sustainability*, **2026**, *18*(2), 778.