www.cadanda.com Computer-Aided Design and Applications

Topological Vision: Applying an Algorithmic Framework for Developing Topological Algorithm of Architectural Concept Design

Chieh-Jen Lin

Tainan University of Technology, t60011@mail.tut.edu.tw

ABSTRACT

This paper applies an algorithmic framework to help architects to model topological algorithms by inputting geometric intentions. By the help of rewritable example scripts and adjustable algorithmic modules, which are based on the previous studies that proposed an algorithmic framework titled STG*f*, this paper applies this framework to help architects to model topological algorithms by inputting geometric intentions. This paper aims to help architects to associate topological knowledge with the algorithmic process of parametric design at early conceptual design stages.

Keywords: parametric architectural design; algorithmic modeling; geometric intention; topological relationship, algorithmic framework.
DOI: https://doi.org/10.14733/cadaps.2019.583-592

1 INTRODUCTION

Parametric design was defined as the exploration of associative relationships among geometric intentions [9]. Beyond generating complex geometries, however, new models of architectural design thinking and strategies also emerge with the popularity of algorithmic modeling tools. As an evolutive result of algorithmic thinking and scripting, Oxman concludes the parametric design thinking is formulated at the intersection of three types of knowledge, which are the cognitive model of architectural design, the process model of digital design, and constructional order of fabrication design [7]. At early and conceptual design stages, which usually do not involve material and constructional requirements yet, the associative relationships between the architectural design knowledge and the algorithmic process of digital design become more critical for parametric architectural design.

Oxman indicated two types of cognitive model of architectural design, which are typological and topological knowledge, and claimed that the topological versions and visions demonstrated a seminal theoretical and operative methodological concept in parametric design thinking [7]. Unfortunately, algorithmic modeling tools lack the assistance for users in associating architectural design knowledge, no matter for typological or topological, with informational processes of geometric intentions. Oxman concluded the knowledge of how to manipulate and explore the associative relationship and dependencies of topological geometries is the critical key to parametric design thinking [7]. However, these kinds of manipulation and exploration must rely on the skills and techniques of algorithmic thinking and scripting. Therefore the scripting and tool-making skills and knowledge seem to be the core of architectural design education and practice when parametric design and algorithmic modeling tools become more popular and important.

One of the reasons for the popularity of algorithmic modeling tools is most of the algorithms, such as mathematical formulas of complex geometries [6], metaheuristic algorithms of multiple-objective optimization [10], the prediction and evaluation formula of building performance [8], were developed and validated in relevant disciplines. However, cognitive researching revealed that designers prefer to apply algorithms only as a means of exploring geometric intentions, but apply known solutions and design patterns for non-geometric intentions [11]. After all, the algorithmic modeling was developed to accelerate 3D modeling tasks by applying algorithms, no wonder designers thus prefer to apply known solutions rather than to develop or implement an algorithm by themselves. However, if the design intentions represented by those known solutions and design patterns, especially the topological design knowledge, can be converted into an algorithmic model, the parametric design should be more useful for exploring non-geometric intentions of parametric design.

Based on the previous studies that proposed an algorithmic framework titled STG*f* [5], this paper applies the framework to help architects to model topological algorithms by inputting geometric intentions. By the help of rewritable example scripts and adjustable algorithmic modules, this paper aims to help architects to associate topological knowledge with the algorithmic process of parametric design.

2 THE APPLICATION OF AN ALGORITHMIC FRAMEWORK

Based on the STG pattern proposed in the previous study [4], which was the semantic-topological-geometric conversion pattern of BIM information schema [1], the STG*f* framework implemented an algorithmic framework by applying Grasshopper and GhPython plugin as an algorithm-aided design tool [5]. By dividing the algorithm of conceptual design into three parts: (1) the semantic module that help architects to indicate the geometric objects and to infer their semantic relation; (2) the topological module that recognizes and validates the topological relations of inputted geometric objects; (3) the geometric module that manipulates and visualizes the topological validations of semantic and topological modules. By providing rewritable example scripts and adjustable topological modules, which are editable clusters of topological algorithms in Python language, this algorithmic framework aims to help architects for developing topological algorithms by inputting their geometric intentions in early architectural design stages.

An algorithmic framework like STG*f* can not only play an design assistant for developing algorithmic models of parametric design, but also provides an approach for converting design knowledge into algorithms. The basic idea of the STG*f* framework implied a loop approach of three developing steps, which are the first definitions of semantic models, then the development of topological controller, and the final validations of geometric views. However, since three modules of algorithms are isolated in the STG*f* framework, it is possible for a user to begin from any one of those modules. As studies indicated, designers prefer to apply algorithms for exploring geometric intention [11]. And it was found the similar approach in previous studies that users usually applied geometric representations of design intentions, which are sketches or diagrams of design concepts, before they retrieved or applied relevant design criteria [3]. Therefore this paper proposes another approach for applying the STG*f* framework to assist designers for converting topological knowledge and criteria from the geometric intentions.

2.1 The Geometric Features as Representation of Design Intentions

The initial purpose of the geometric module in the STG*f* framework was provided to demonstrate how to input geometric objects from Rhino into the semantic and topological modules, then to provide visual clues for the validation of users' design intentions. For users who have sketches or diagrams of design concepts, the geometric module of the STG*f* framework can apply to help user for retrieving design knowledge based on the geometric features.

The sketches or diagrams of design concepts are a general means for architects to representing and communicating their intentions. The traditional approach for retrieving the design knowledge within this kind of materials usually applies protocol analysis of designers' recalling. Thanks for the powerful processing ability of modern tools, nowadays it is very easy for users to photo or scan their sketches or diagrams into modeling tools like Rhino. However, one propose of this approach is not only for converting the 2D analogic images into 2D/3D digital geometries, but also provide the computable geometric representations of design intentions. Since designers prefer to apply known solutions and design patterns for other non-geometric issues [11], it needs further steps to analyze then retrieve non-geometric intentions and design knowledge from these representations.

2.2 The Semantic Ontology as Parametric Schema of Design Intentions

At early design stages, design intentions usually consist of abstract, textual descriptions concerning various design objects and their relationships. Although essential semantic information regarding building components has been predefined in BIM and Industry Foundation Classes (IFC) schema, however, designer's intentions will not be limited in the domain of those schemas. On one hand, Rhino has no predefined semantic schema of building information, not to mention the definition outside the BIM or IFC. On another hand, the free and open definition of a semantic schema may encounter conflict issues of similar identity names. For allowing architects to define or interpret their unique objects and relationships, it needs a contextual semantic ontology of design intentions [3].

Semantic ontology is a computational format for representing, storing, and validating a semantic ontology of domain knowledge. The initial purpose of the semantic module in the STG*f* framework was to apply the ontology technique for helping designers to capture the semantic logic of abstract design knowledge and intentions in order to establish the parametric schema of validating algorithms. By hooking the inputted geometric features with the defined semantic ontology of design knowledge and intentions, the STG*f* framework can help architects associate their abstract design knowledge with geometric intentions in order to develop topological controlling algorithms for exploring non-geometric intentions.

2.3 The Topological Algorithm as Generative Controllers of Design Intentions

Topology is the mathematical relations among different objects, therefore is the critical information for validating the conceptual consistency of design intentions. Because there is no unanimous definition of necessary topological information in AEC domain, both BIM and IFC schema ignores most of the topological information among different building components [1]. Especially in the early stages, however, architects usually care more about spatial topologies, such as adjacent, overlapping, surrounded, separated, et al. [2], than other kind of mathematical relationships. Therefore, unlike generative algorithms focuses on generating geometric forms, there should be at least two kinds of topological algorithms, which are validating and adapting algorithms for a given spatial topology. A validating algorithm should be able to validate whether the inputted objects meet a given topology or not. An adapting algorithm should be able to modify the inputted objects in order to satisfy a topology.

Since it needs more investigations for how to automatically modify inputted objects in order to satisfy a given topology, especially for the topology that involves more than two objects will face technological challenges. Therefore the initial function of the topological module in the STG*f* framework only provides the example scripts of validating algorithms for basic spatial topologies. By visualizing the validating results based on defined semantic ontologies and inputted geometric features, this paper proposes the topological vision as an assistant for helping architects to develop algorithmic modeling of parametric architectural design.

3 INITIAL TESTING AND EVALUATION OF TOPOLOGICAL VISION

In the previous studies, three exams of Taiwan's architect qualification in recent years had been studied for encoding abstract design concepts in the early design stages. Since the contexts of three given sites had no geometric features for retrieving relevant design criteria, there had no obvious clues that would constrain candidates' geometric intentions. Except for the inner contexts of the given site, such as existing trees, however, the exam in 2017 is "a community center on a historic street", and provided explicit geometric contexts next to the given site (Fig. 1a). There is a temple of the local land god located at the east street opposite (Fig. 1b), a row of baroque style, one-story, classic street houses located at the west street of the site (Fig. 1c), and a traditional market located at the west street opposite (Fig. 1d). The geometric features of the site's contexts therefore not only provide clues for retrieving design criteria of building forms, but also restrict the candidates' geometric intentions.

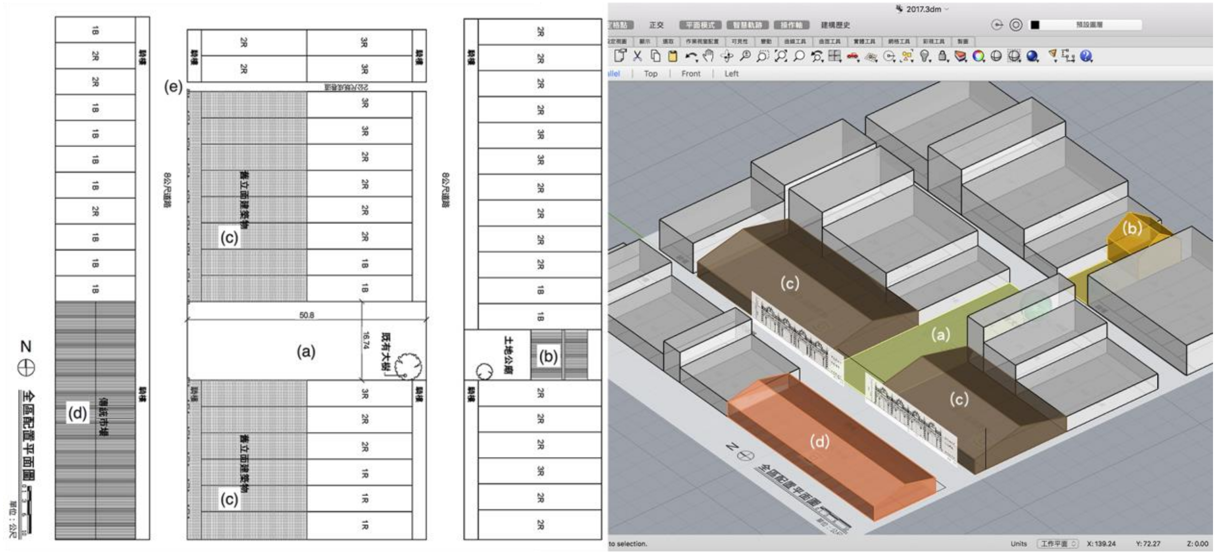

Figure 1: The site's contexts of the architect qualification exam in Taiwan 2017: (a) the east and west side of the site are adjacent to an eight-meter width street, (b) a temple of the local land god locates on the east street opposite, (c) a row of baroque style street houses located at the west street are adjacent to the site, (d) a traditional market locates on west street opposite, and (e) a 2 meters lane at north.

3.1 Example of Geometric Contexts

Unlike the site's contexts in the previous exams without geometric features, the exam in Taiwan 2017 provides the façades of the baroque style street houses (Fig. 2). The one-story baroque style façade of a street house unit is 5.58 meters wide and 5.93 meters high. The arcade is 3.82 meters height with four columns and three round arches, and the parapet above the arcade is 2.11 meters height with baroque style decorations. Those geometric features imply the façade design along the historic street should reflect the basic geometric patterns, which are 5.58 meters grids on the plan, and 3.82 meters level on the first story of the elevation of the community center.

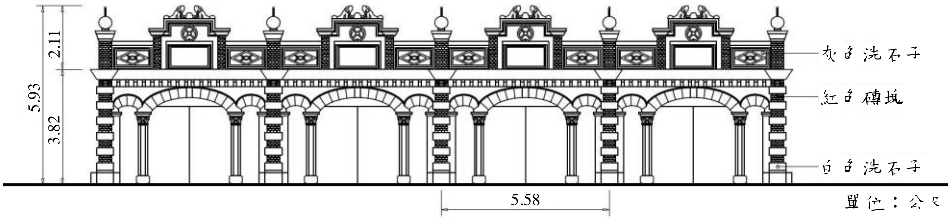

Figure 2: The geometric features of the baroque style street houses in the site's contexts of the architect qualification exam in Taiwan 2017.

The width of the site's frontage is 16.74 meters, which is exactly the width of three façade units. It is obviously implies that the façade design of the new community center should filled three façade units on the street in order to connect the existing arcade and to recover the historical streetscape. However, this obvious clue may also be a trap of the exam. Candidates may ignore the relationship between the temples and the market on both sides of the site. The temple and the market are important activity nodes for everyday lives of the community. Therefore, these two nodes suggest an unobtrusive but important community activity, which may relay the 2 meters width lane located at the north of the site (Fig. 2e). When a candidate try to recover the arcade of the historical streetscape, it mays also block the possible connection between the temple and market. In this exam, however, how to create the possible connection of everyday lives is exactly what a community center requires.

3.2 Developing Topological Vision from Geometric Intentions

The building program of the 2017 exam consists of a multifunctional hall for 130 people meeting, a library, an office, and a parking lot for two cars and some motorcycles. As the "community-friendliness" subjects in previous three years exams, however, the critical design issues of the exams are not only related to the relation among the interior spaces of the building, but also related to the correlations between the community and the building. These correlations, in other words, is about to how to arrange the building in order to shape the public spaces, such as the arcades along two streets, and to connect the open spaces, such as the yard in front of the temple for the community activities. The outdoor spaces shaped by a building and its surroundings are usually ignored in the information schema of BIM or IFC. In addition, what topological relations among the geometric features of a community center and its existing contexts can facilitate community activities still can leave much room for architects to interpret. Therefore, the candidates must try to not only design the building, the public and open spaces within the site, but also must explain how the building and those spaces can facilitate community activities.

For the "community-friendliness" subject of exam 2017, the candidates must try to develop spatial connecting between the traditional market and the temple. However, only the connections of public and open spaces, such as arcades, public parks and plazas, can facilitate the community activities between the traditional market and the temple. Therefore, the basic design concept should be to connect the traditional market and the temple with a serial of public spaces, such as arcades, public parks and plazas. Therefore, this concept needs at least two algorithms: (1) to identify an object is a public space or not, (2) to recognize two nodes are connected by public spaces. The first is about semantics of a design object; the second is about the topologies among design objects.

Although a good public space should have specific topological correlation with other indoor/outdoor spaces, however, it usually has no specific geometric features, which are depended on the architect's creativity. Therefore, the easiest way to identify a public space is to assign semantics to geometries. For example, a user can apply the "Semantic" module of STG*f* to indicate the sub classes of "Public Space," such as "Arcade," "Park," and "Street," and indicate the semantic feature of Rhino's geometries, such as "Layer Name." Then, STG*f* can infer geometries within

"Arcade," "Park," or "Street" layer as a kind of "PublicSpace." This criterion can also be represented in semantic web rule language (SWRL) as Eqn. 1.

$$(\text{Arcade} \vee \text{Park} \vee \text{Street})(?x) \rightarrow \text{PublicSpace}(?x) \tag{1}$$

Secondly, it needs a "Connecting" topology to recognize whether two spaces are connected by other public spaces. Fortunately, the "Adjacent" and "Overlap" topology are two basic topological algorithms of the "Topology" module in STG*f*. Then, a user can establish a "Connecting" topology by apple the "Adjacent" and "Overlap" topology of STG*f* (Eqn. 2). Base on two algorithms above, the user can establish a "PublicConnecting" criterion to recognize whether two spaces are connecting by public spaces (Eqn. 3). Based the semantic algorithms above, a user can validate whether two design objects, such as the market and the temple around the site, are connected by public spaces or not.

$$\text{Adjacent}(?x, ?y) \vee \text{Overlap}(?x, ?y) \rightarrow \text{Connecting}(?x, ?y) \tag{2}$$

$$\text{Connecting}(?x, ?z) \circ \text{Connecting}(?y, ?z) \circ \text{PublicSpace}(?z) \rightarrow \text{PublicConnecting}(?x, ?y) \tag{3}$$

3.3 Implementation of Topological Visions

One reason for why the BIM or IFC schema ignores the outdoor and open spaces is because of their geometric features are ambiguous. An outdoor/open space is sometimes just the left space after buildings occupy the site. Especially when there are natural landscapes, such as trees, waters, or slopes in the site, architects sometimes would try hard to reduce artifacts such as hard paving. Architects usually apply the building massing to shape, rather than directly to draw a shape of an open space. In other words, the geometric features of open spaces usually are determined by the buildings within and surrounding the site. For validating the public connecting concept of two community nodes, it needs to implement the geometric algorithm for generating the ambiguous open space.

Since the building program of the exam 2017 consists of a hall, a library, a parking lot, and an arcade, a user can indicate them as the sub classes of "Building" in the site (Eqn. 4). Consequently, the open space of the community center in the exam 2017 is the site's open space, which should be equal to the site minus the buildings (Eqn. 5). By applying the "minus" algorithm of "Geometric" module in STG*f*, the algorithm can generate the open space of the community center by insert geometries into the "Site," "Acrade," "Hall," "Library," and "Parking" layer in Rhino. By collecting the generated geometries of open spaces into "Park" layer in Rhino, the STG*f* can generate the topological vision of public connecting between the temple and the market of the exam 2017 in Taiwan. Even though the user dose not "draw" a open space as the park of the community center, the algorithm can high light the open space shaped by the arcade and the building to suggest a potential proposal for facilitating community activities (Fig. 3).

$$\text{Hall}(?x) \vee \text{Library}(?x) \vee \text{Office}(?x) \vee \text{Parking}(?x) \vee \text{Arcade}(?x) \rightarrow \text{Building}(?x) \tag{4}$$

$$\text{Site}(?s) \circ \text{Building}(?x) \circ \text{In}(?x, ?s) \rightarrow \text{OpenSpece}(?s, \ \text{minus}(?s, ?x)) \tag{5}$$

Another important requirement of the exam 2017 is to maintain or to recover the classic arcade of the historical streetscape. The basic idea therefore is to fill a new arcade on the west street of the site to connect two existing arcades. For validating this idea, a user can extend the previous algorithm to ensure the historical streetscape recovered (Eqn. 6). In addition to the "Connecting" topology, the critical criterion for recovering the historical streetscape is to match geometric features of the existing baroque style façade.

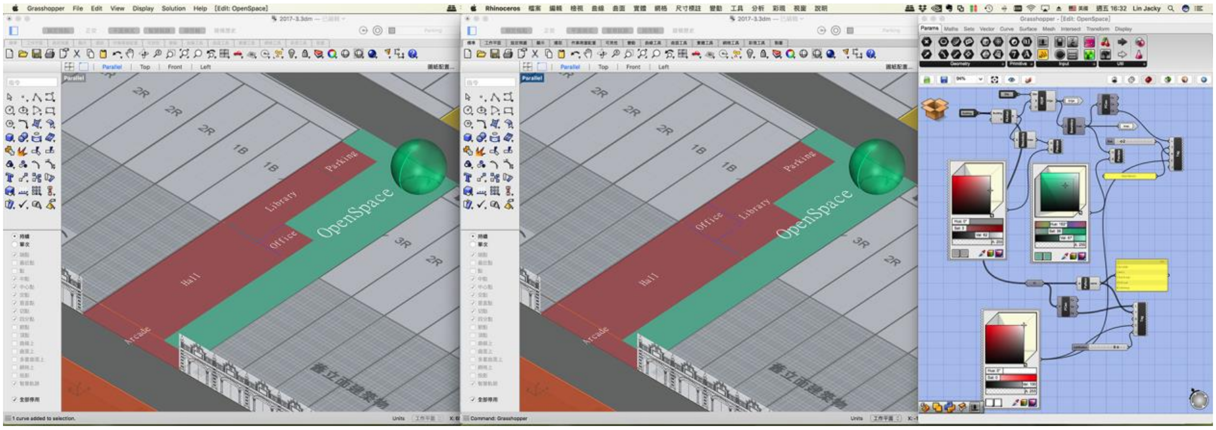

Figure 3: Extending the "minus" function of the "Geometric" module in STG*f* to generate the "OpenSpace" of the community center by inputting two different plan layouts of the building's areas.

At the early and conceptual design stage, however, the geometric intention may only focus on the matching of the façade modulus, that is the 5.58 meters by 3.82 or 2.11 meters modulus, rather than the geometric details of the baroque decorations. Thanks to the powerful Grasshopper, it is easy for users to generate the new arcade by inputting the geometric features of the baroque style façade. For example, a user can input the image of a façade unit for visualize the modular concept for recovering the historical streetscape (Fig. 4).

$$\text{Connecting}(?x, ?z) \circ \text{Connecting}(?y, ?z) \circ \text{Acrade}(?x) \circ \text{Acrade}(?y) \circ \text{Acrade}(?z) \circ \text{Site}(?s) \circ \text{In}(?s, ?z) \rightarrow \text{RecoverdAcrade}(?x, ?y) \qquad (6)$$

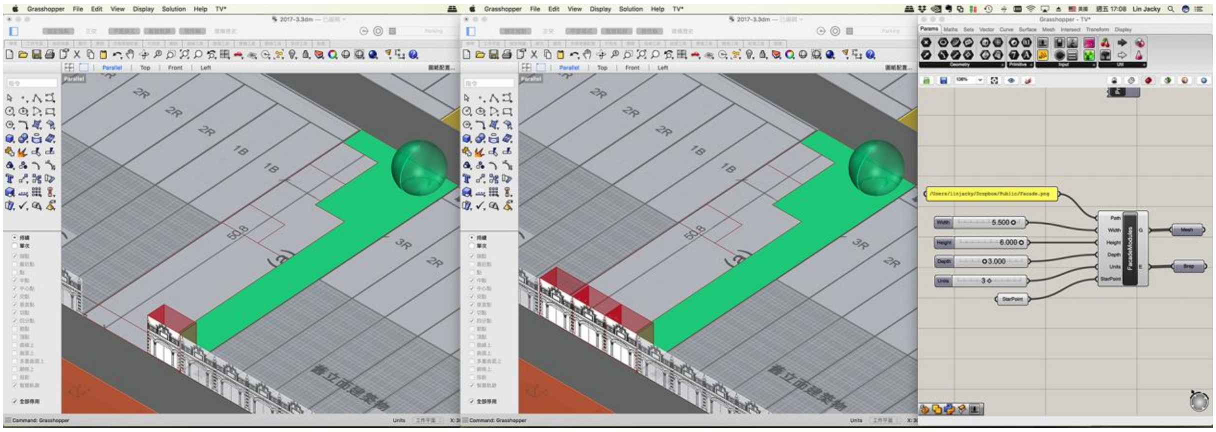

Figure 4: Applying the "Geometric" module of STG*f* to generate the units of the baroque style façade: (a) firstly to input the start point and insert the façade's image in Rhino (Left), and then (b) to increase the number of façade's units in Grasshopper (Right).

4 DISCUSSION

Computational architectural design applies algorithms to solve architectural design problems, but should not be limited to creating geometries of building's form. After all, the building's form should be the means for solving problems, not the problems themselves or the only purpose. As Oxman claims that the topological knowledge of architecture is a critical key to parametric design thinking, how to embedding the topological knowledge of architectural design into the generating algorithms is also the critical key for validating weather a architectural design problem is solved or not. Based on this view of computational architectural design, to apply the STG*f* framework for developing topological algorithms is discussed as follows.

4.1 Ambiguous Semantics of Geometric Intentions

Even though a geometric intention is explicit, however, the semantic criterion for this geometry may still be ambiguous at early design stage. For example, the building program of the community center in the exam 2017 in Taiwan requires a "multifunctional hall" for the community meeting and other usages. In Taiwan, a community center often holds various activities, such as wedding parties, recreational performing, competitions, election rallies, and so on. Therefore it implies that it does not need a full-enclosed interior space to meet the soundproofing requirements. In this case, a hall with enough openness can also be regarded as an open and public space like the arcade on the west street and the temple's square. Then, an opening hall can be filled into the serials of public/open spaces between the market and the temple, and it does not need to leave an open space for connecting the arcade and the temple's square as the open space criterion mentioned above (Fig. 4).

Although it is easier to indicate the opening "MultifunctionalHall" as a kind of "PublicSpace" like the "Arcade" class in the "Semantic" module of STG*f*, However, it needs further design criteria to validate the accessibility of this hall. For example, the semantics of "Openness" or "EnclosedRate" have been studied in previous studies [4], therefore the "HighOpenness" criterion or the "LowEnclosedRate" of walls can be used to validate the accessibility of a "MultifunctionalHall." Since the details of walls to enclose a spaces usually are still abstract at the early design stage, and the validation of "HighOpenness" or "LowEnclosedRate" needs more parameters and computing power, the validation by explicit semantics of the "Semantic" module in STG*f* is easier to be understood and implemented by users.

4.2 Semantic Reasoning of Topological Relations

Since the "Topology" module of STG*f* has the "Separated" topology (Fig. 1), then it seems to be easier to apply the negated "Separated" topology (Eqn. 7) for implement the criterion of the "Connecting" topology as Eqn. 2. However, the monotonicity of SWRL does not support the negated atom reasoning, and the SWRL reasoner cannot accept a simple negated rule like the Eqn. 7. Therefore, the semantic rule of "Connecting" topology needs apply the monotonic "Adjacent" and "Overlap" topology to implement in the "Semantic" module for apply the SWRL reasoner. However, thank to the full logic function of Python language, it is easy to script a negated rule in GhPython component to implement negated atom reasoning. This technique can overcome the monotonicity of the SWRL reasoned, but it needs more programming skills for implementing.

$$\neg\ \text{Separated}(?x, ?y) \rightarrow \text{Connecting}(?x, ?y) \tag{7}$$

To manipulate the topological relations is critical for the architectural design at the early stage. The semantic reasoning is not only useful for quickly validating design criteria, but also easier to be understood by and communicated with stakeholders of a building project. As the algorithms become more and more complex in parametric design, to immediately and visually prompt the semantic validations of the topological relations may be the easiest means to discover weather an algorithm is satisfied the design criteria or not (Fig. 1).

4.3 Topological Visions of Geometric Intentions

Sketches and diagrams are the traditional means for representing the design concepts for these kinds of topological relations among the geometric features of the design proposal. The design concepts of sketches or diagrams usually only represent what a design criterion is, rather than how the criterion can be validated. Therefore, those representations actually represent the semantic ontologies of topological criteria, rather than validating algorithms of topological relations. However, sketches and diagrams of design concepts can not only provide semantic parameters of validating algorithms, but also the visual clues of validating algorithms. By inputting those geometric features of design contexts into the STG*f* framework, such as the images of street house's facades, the shapes of existing trees, and the temple's square, the topological vision aims to help architects for exploring topological relations among given geometric features of the design contexts at early stages. By visualizing the topological validations, such as the connections of a serial of public/open space between the market and the temple in the examp2017 in Taiwan, the topological vision can help the candidates to explore their design concepts and to avoid the trap for breaking the public/open spaces' connections.

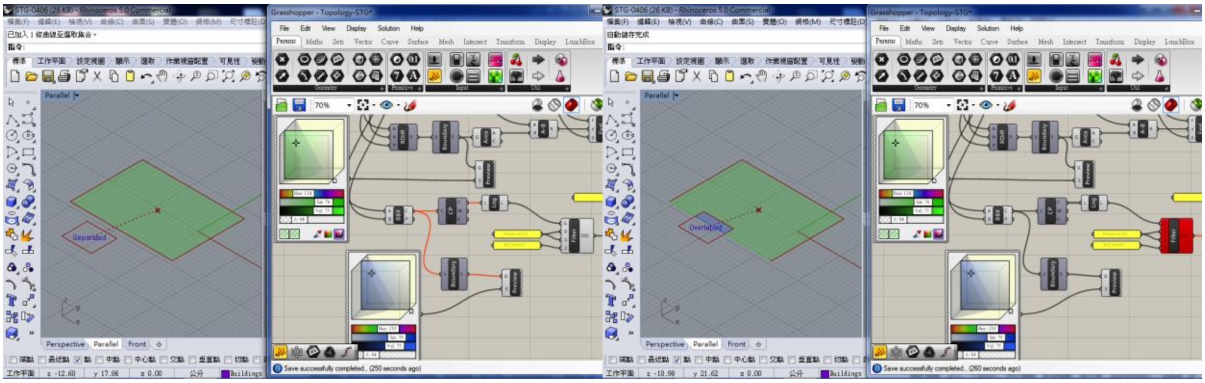

Figure 5: A demonstration of topological vision from geometric intentions.

5 CONCLUSIONS

As mentioned above, designers prefer to apply known solutions and design patterns for non-geometric intentions. The STG*f* aims to help architects to convert the topological knowledge within known solutions and design patterns into an algorithmic model for exploring more possible solutions. By providing rewritable example scripts and adjustable algorithmic modules, which are editable clusters of algorithmic components in Grasshopper, this paper aims to help architects to associate topological knowledge with the algorithmic process of parametric design.

ACKNOWLEDGEMENTS

The Ministry of Science and Technology of Taiwan supported this paper under grant number MOST 106-2221-E-165-001-MY2.

Chieh-Jen Lin, http://orcid.org/0000-0001-9981-5864

REFERENCES

[1] Eastman, C.; Teicholz, P.; Sacks, R.; Liston, K.: BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors, 2nd ed., John Wiley & Sons Inc., Hoboken, N.J., 2011. https://doi.org/10.1002/9780470261309

[2] Lin, C.-J.: Architectural Knowledge Modeling: Ontology-Based Modeling of Architectural Topology with the Assistance of an Architectural Case Library, Computer-Aided Design and Applications, 12(4), 2015, 497-506. https://doi.org/10.1080/16864360.2014.997647

[3] Lin, C.-J.: Design Criteria Modeling - Use of Ontology-Based Algorithmic Modeling to Represent Architectural Design Criteria at the Conceptual Design Stage, Computer-Aided Design and Applications, 13(4), 2016, 549-557. https://doi.org/10.1080/16864360.2015.1131551
[4] Lin, C.-J.: The STG pattern – application of a "Semantic-Topological-Geometric" information conversion pattern to knowledge-based modeling in architectural conceptual design, Computer-Aided Design and Applications, 14(3), 2017, 313-323. https://doi.org/10.1080/16864360.2016.1240452
[5] Lin, C.-J.: STG-framework: A Pattern-Based Algorithmic Framework for Developing Generative Model of Parametric Architectural Design at Conceptual Design Stage, in: Proceedings of the CAD Conference and Exhibition 2017 (CAD'17), CAD Solutions, Okayama, 2017, 288-292. https://doi.org/10.14733/cadconfP.2017.288-292
[6] Miller, N.: Parametric Strategies in Civic Architecture Design, in: Proceedings of the 29th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA 2009), Chicago 2009, 144-152.
[7] Oxman, R.: Thinking difference: Theories and models of parametric design thinking, Design Studies, 52, 2017, 4-39. https://doi.org/10.1016/j.destud.2017.06.001
[8] Roudsari, M.S.; Pak, M.: Ladybug: A Parametric Environmental Plugin for Grasshopper to Help Designers Create an Environmentally-Conscious Design, in: Proceedings of the 13th International conference of Building Performance Simulation Association, Chambery, France, 2013.
[9] Woodbury, R.: Elements of Parametric Design, Routledge, New York, 2010.
[10] Wortmann, T.; Nannicini, G.: Black-Box Optimisation Methods for Architectural Design, in: 21st International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA 2016), Melbourne, 2016, 177-186.
[11] Yu, R.; Gero, J.; Gu, N.: Architects' Cognitive Behaviour in Parametric Design, International Journal of Architectural Computing, 13(1), 2015, 83-102. https://doi.org/10.1260/1478-0771.13.1.83