DESIGN CRITERIA MODELING

Ontology-based parametric modeling for representing architectural design criteria at the conceptual design stage

CHIEH-JEN LIN
Tainan University of Technology, Tainan, Taiwan
t60011@mail.tut.edu.tw

Abstract. This paper proposed an ontology-based parametric modeling tool, “Design Criteria Modeling (DCM),” which applies a graphic predicative tool and semantic ontologies of architectural topology. DCM was intended to help architects in representing, exploring, and validating design criteria with parametric 3D model at the early design stage. By applying a reasoner of semantic ontology, architects could use DCM to determine whether conceptual models meet the semantic ontology of proposed design criteria.

Keywords. Architectural information modeling; architectural design criteria; semantic ontology; parametric design.

1. Introduction

The 2D drawings of CAD has been replaced with BIM by 3D visualization as a means of solving information integration problems among different disciplines within the AEC industry. However, the heavy cognitive and learning workload caused by the complexity of BIM applications remained a major obstacle in architectural design. According to suggestions for the levels of development (LOD) protocol (Bedrick, 2013), lower LOD objects, such as massing objects in Autodesk Revit, can reduce the burden of modelling works at the early and conceptual stages by minimizing details. However, since BIM focuses on representing a building product, rather than the design processes, it is difficult to present and deliver design criteria for generating or validating within a 3D model.
For early and conceptual design, rapid modeling tools, such as SketchUp and Rhino, have been increasingly used by architects for freely exploring geometric possibilities rather than BIM applications, which are more useful for detailed modeling. But rapid 3D models lack semantic information about the fundamental components of a building, and it is difficult to determine whether the models satisfy proposed design criteria. Since generative modeling tools like Grasshopper emerged, generating forms by means of algorithms as representations of design criteria has become much easier than before. Algorithm-based generative modeling has therefore become popular among architects for generating complex building forms when engaging in geometric creativity during early design stages. However, except in the case of abstruse issues which other design criteria can be communicated by generative algorithms, how to compose algorithms in order to meet the requirements of design criteria are still more technically challenging than pure design creativity.

To assist architects in representing, exploring, and validating design criteria using parametric 3D modelling at an early stage, this paper proposed an ontology-based parametric modeling tool named “Design Criteria Modeling (DCM),” which applied the results of previous projects, including visual ontologies of architectural design (Lin, 2013) and a predicative tool for architectural topology (Lin, 2014) based on the OWL of Protégé\(^1\). The goal of DCM was to integrate architectural design knowledge with parametric design tools.

2. The Approach of Design Criteria Modeling

In the early and conceptual design stages, architectural design does not only involve representing and validating known design criteria, but also learning and developing new criteria. However, not all conceptual design criteria are visible and obvious in 3D models. For examples, minimal space requirements (Banks, 2014), visual field and privacy issues of openings, circulation of indoor and outdoor spaces, and other design issues proposed by architects are different from issues of concern in the engineering and construction disciplines. Traditionally, architects employ other visual media, which usually consist of a series of diagrams or precedents able to represent architects’ beliefs and intentions. Since those visual media are separate from 2D drawing and 3D models, their information cannot be easily converted into BIM or other applications for validation. However, with the emergence of algorithm-based generative modeling tools such as Grasshopper, a graphic programming interface has become a promising method for representing geometric criteria about building forms.
Nevertheless, the absence of appropriate annotations sometimes leads to communication difficulties when working with graphic compositions of algorithms, especially for abstract criteria other than geometric features. BIM design information consists of three types of design information—geometric, topological, and geometric—and architectural design can be considered as the conversion and processing of these three types of design information. This paper therefore proposed a “geometric-topological-geometric” conversion framework for modeling conceptual design criteria at an early stage of the design process.

2.1. MODELING ONTOLOGY OF SEMANTIC CRITERIA

Architectural conceptual design always begins with the elaboration of building program, site, and other design contexts, which usually consist of textual descriptions about various requirements. The primary design criteria are consequently the semantic elaborations of design objects and their objectives. In BIM, semantic information about building components is default and self-evident. However, the semantic criteria proposed by architects may not only represent the consensus regarding the design context, but also the epistemology about how to interpret the design context. They require further validation, in order to avoid contradictions. Semantic ontology techniques, therefore, were incorporated into the DCM, in order to meet the semantic criteria.

Semantic ontology is a knowledge representation technique in artificial intelligence for developing a semantic web (Antoniou and Harmelen, 2004), which can enable a search engine to understand users’ intentions. Protégé is a popular tool for developing a semantic ontology of domain knowledge, and a logic reasoner based on semantic web rule language (SWRL) can help knowledge engineers to validate and infer implicit knowledge within an ontology. Therefore a SWRL reasoner was applied to DCM as a tool for validating proposed criteria. However, DCM was not intended for developing a complete ontology of architectural knowledge, but rather providing a rapid mechanism for representing and validating partial ontology of proposed design criteria, for example, assuming a set of semantic design criteria as follows: (1) “HabitableRoom” is a kind of “IndoorSpace” with “hasWindow” property, and (2) “UninhabitableRoom” is an “IndoorSpace” without “hasWindow,” (3) “UninhabitableRoom” is not a “HabitableRoom.” The logic reasoner can infer an “UninhabitableRoom” contradiction when a space is claimed to be a “HabitableRoom” but without any window. Although the semantic logic reasoner cannot automatically deduce solutions for satisfying the proposed design criteria, it can detect contradictions, and can help architects maintain the consistency of design criteria.
2.2. MODELING RELATIONS OF TOPOLOGICAL CRITERIA

Topologies are the mathematical connections of components in BIM and the fundamental definitions of parametric modeling (Eastman et al., 2011). However, the topologies of BIM applications are prerequisite knowledge, which is embedded in the parameters of models for the purpose of fabrication and construction, and cannot be freely manipulated by architects. Topological relations nevertheless also constitutes critical information used for validating whether a model satisfies the semantic ontology of proposed design criteria. For example, whether a space is a “HabitableRoom” depends not only on whether it has a “Window,” but also on the orientation of the “Window” “FaceTo” outdoors space for effective ventilation and day lighting.

By applying generative modeling tools like Grasshopper, simple topologies between two geometries, such as separated, connective, adjacent, and enclosing, are more easily detected than ever. However, complex topologies about multiple objects, such as surrounding, centralizing, clustering, and branching (Ho and Wang, 2009) that were easily recognized by humans or generated by algorithms occasionally could not be easily detected or validated by algorithms. As a prototype modeling tool, the DCM focused on providing conversion algorithms for associating simple topologies about two objects with semantic relations in the ontologies of design criteria.

2.3. MODELING FEATURES OF GEOMETRIC CRITERIA

Geometric features are the primary parameters of BIM components, and modeling the detailed geometric features of building components is therefore one of the major tasks of the BIM applications. In the early design stage, however, design objects are usually abstract, and their geometric features are simplified, in order to speed up the validation of the topological relations among objects. By applying Grasshopper as an algorithmic modeling tool, the DCM was therefore able to use the geometric functions of Rhino for modeling the geometric features of design objects.

Since generative modeling tools like Grasshopper can input geometric objects in Rhino as parameters for generating 3D models, the DCM therefore also took the geometric features as conceptual design objects, such as curves, 2D shapes, and 3D massing, as input parameters, and then generated validation results by the means of algorithms. However, unlike generative modeling, commonly used to generate complex and sophisticated 3D models, the DCM first attempted to model invisible or non-obvious design criteria, such as minimum space requirements, view fields of openings, and geometric constraints of the building code on design objects, rather than automatically generate optimal or possible solutions.
2.4. SUMMARY

Based on the foregoing “geometric-topological-geometric” information conversion framework, the DCM consequently divided conceptual design tasks into three modeling steps: (1) semantic ontologies of design objects or objectives, (2) topological relations among those objects, and (3) geometric features for visually validating ontologies by their topologies. The DCM was not only intended to provide a tool for incorporating design criteria, especially for invisible or non-obvious in 3D visualization of parametric models, as well as guidance for architects in explicitly presenting their design beliefs and intentions.

3. Implementation and Initial Evaluation of DCM

One of initial motivations for developing the DCM was to improve the knowledge representation abilities of Grasshopper in architectural design, and the initial prototype of DCM was developed within Grasshopper. Since Grasshopper is a DotNET plugin for Rhino, then VisualBasic or C# should be the first choice of programming language for developing plugins for Grasshopper. However, the integration of semantic ontology techniques was the biggest difference between the DCM and other Grasshopper plugins. Since Protégé was developed in Java, most ontology tools were available only in Java. Fortunately, the new Rhino 5 also supported the Python script language, and some Python libraries, such as RDFLib and FuXi, supported the manipulation functions of RDF/OWL and logical reasoning of SWRL, and Jython support Python script in a Java virtual machine (JVM). The GhPython plugin, providing a hook for Python scripts for Grasshopper, therefore became an ideal tool for implementing the DCM.

The DCM prototype was tested by students. They were asked to rapidly design an “Architect’s Office as a Good Neighbor,” a topic on the architecture exam of Taiwan in 2014. The urban context of the site included mid-rise housing in the east, several low-rise houses in the west, a community park in the south, and a primary school in the north. In addition, the site also contained two old trees and an old two-story dormitory for public servants, and the trees and old dormitory must be retained and reused (Figure 1.a.). Except for basic issues of the building code, traffic, and climate response, the design criteria of this project involved how the design context would affect the massing of the office building, and how the indoor spaces in the office building could be arranged so as to be a good neighbor to the community. Through the guidance of the DCM, adjacent houses, the dormitory building, the old trees, and contextual objects were the primary parameters of algorithms for modeling design criteria.
The contexts and contents of the site actually constrain and suggest feasible layouts of the building massing and interior spaces. While these clues were possibly not obvious, or even invisible for novice students, how to express the influence of design context was the major purpose of the exercise. Experienced students knew that the two old trees constituted favorable conditions for shaping outdoor spaces, providing scenery and ventilation for indoor spaces. Most of the students therefore chose to minimize changes on the south side of the site, which was close to the community park. They would like to utilize the park for neighborhood events. They also located the building massing, in order to avoid damage to the old trees and the dormitory. However, the strong sunshine from the west was a climate issue. Experienced students recognized this consequence of their choices, and subsequently took appropriate countermeasures, such as deep balconies, vertical shadings, double walls, plantings, and water pools. In this case, students were asked to model the western trees as an input parameter for design criteria. This reminded novice students to recognize whether it was a beneficial condition or not, and then propose appropriate criteria for how the building should respond to it.

4. Discussions

In order to enhance the utility of generative modeling tools such as Grasshopper in representing and validating design criteria during early architectural stages, this paper proposed a “geometric-topological-geometric” conversion framework for modeling design criteria via 3D visualization. This information-driven approach is discussed as follows.
4.1. SEMANTIC ONTOLOGY OF DESIGN CRITERIA

Primary design criteria were described using textual information, in order to establish reasoning logic about how to validate whether the proposal was satisfied or not. Unlike the algorithms in Grasshopper, which directly generated geometric results, the logic within the semantic ontologies of design criteria only judged whether a proposal satisfied certain criteria. In the previous architectural exercise, for example, if the two old trees were taken as a “necessary condition” in the ontology, then the DCM would declare that the deletion or relocation of the trees was inconsistent with this criterion. The missing associations between semantic criteria and a proposed design frequently became the causes of disagreements in both design education and practice. The DCM could promote communications among stakeholders of a project, as in the case of students playing architects and design teachers playing clients and consultants in an architectural school.

By employing an ontology editor like Protégé, a semantic ontology of design criteria could be validated in advance, avoiding inconsistencies. However, the CDM did not implement full functions of ontology techniques, and the ontologies of conceptual design were partial and incomplete even though inconsistencies and contradictions among design criteria were common and inevitable in early architectural design. In the previous architectural exercise, for example, the requirement for western sun shading in response to climate issues and the utilization of the old trees for scenic and space purposes during neighborhood events reflected a contradiction about openings in the building massing on the western side. But while the problems involving design criteria might cause trouble to AI knowledge engineers, they also served as drivers of architectural creativity. When encountering these problems, students did not attempt to modify criteria to avoid contradictions, but also to take the challenges.

4.2. TOPOLOGICAL RELATIONS OF DESIGN CRITERIA

Converting semantic design criteria into generative algorithms of geometric features was the first difficult part that novices encountered while applying generative modeling tools to design criteria at an early design stage. Topological relations among design objects therefore provided clues for selecting input parameters and composing algorithms. Using the basic definitions of topological relations, including separated, connective, adjacent, and enclosing, the DCM could assist users in composing more topological relations associated with the semantic ontologies of design criteria. In the previous architectural exercise, for example, the idea of shaping outdoor spaces using two old trees were converted into an enclosing topology of the building...
around the trees, or a separating topology between trees and building as a minimal void of outdoor spaces.

Topological relations constitute critical information used in validating the semantic ontology of design criteria, but it was difficult for architects with insufficient programming skills to implement their ideas using this approach. By applying the algorithms of Grasshopper, with the DCM, users were able to easily validate the design criteria of topological relations. Although it was possible to use the DCM to automatically generate possible design criteria solutions in Grasshopper, the DCM was more useful in the visualization of invisible or non-obvious design criteria than in generating forms by means of algorithms. In the previous architectural exercise, for example, if the two old trees and the community park were modeled as “must-see” scenery from interior spaces, it was not necessary to explain why there were western windows facing them.

4.3. GEOMETRIC FEATURES OF DESIGN CRITERIA

The ability to easily generate sophisticated forms through the use of algorithms was the biggest advantage of generative modeling tools like Grasshopper. However, abstract architectural design criteria could not easily be converted into the parameters of algorithmic components in Grasshopper. The geometric features of design objects, associated with semantic ontology and defined topological algorithms in the DCM, could therefore not only be applied to the representation of design criteria, but also to the input parameters of algorithms. In the previous architectural exercise, for example, different shapes of objects were taken as the input parameters of topological algorithms in the DCM. Students inputted a circle, cylinder, sphere, or other irregular shape as “tree” parameters representing retained criteria, and the validation results obtained by the DCM varied with the geometric parameters.

The modeling of design criteria therefore not only involved modeling the design building itself, but also other invisible and non-obvious contextual design aspects as recognized by architects that would shape the building. Geometric features of the design context could affect the expression of design criteria. However, since the DCM was not for generating optimal results, the results of validations in the DCM retained sufficient room for geometric creativity.

5. Conclusions

One of the major complaints regarding the BIM applications was their complexity, causing cognitive overload and learning difficulties (Eastman et al,
For conceptual development and schematic design, algorithm-based generative modeling tools such as Grasshopper have become popular among architects because of their ability to handle complex geometric forms through the simple graphic combination of algorithmic components. However, as the generative algorithms in Grasshopper become more complex, even experienced architects may have difficulty remembering why specific algorithms have certain compositions, much less try to explain or communicate their ideas with others. However, since not all issues could be easily represented and discovered by product-oriented 3D models, proposals have been made to attach more informational dimensions to the 3D BIM, such as time or phasing (4D), cost (5D), energy performance (6D), and facility management (7D) (Banks, 2012). Architects will continue to apply 2D sketches and diagrams when studying and solving design issues in early design stages, and not all architectural design issues require a 3D model for representation or validation. For example, minimum space requirements are more easily recognized in 2D drawings. And the efforts to integrate more design issues into BIM may be caught in the dilemma of an infinite D’s model.

Some design criteria may be too axiomatic, and thus need not be indicated, such as minimum space requirements, sufficient ventilation, daylighting, and ceiling height in a habitable room. Other criteria may be too personal to be integrated, such as the epistemology of design contexts, formal aesthetics, and stylistic preferences. These criteria are inevitably lost and not necessarily included in the BIM. Since 3D models generated using the BIM have become a powerful medium of communication in the AEC industry, how to represent and validate proposed criteria, especially those that are invisible or non-obvious in 3D models, and to communicate ideas involving those criteria with other stakeholders and disciplines at an early stage, is still a critical issue for architectural design. The DCM approach is described above as a first effort to apply Grasshopper as a generative modeling tool in modeling design criteria, which can be associated with abstract semantic ontologies and topological relations. However, how to accelerate and simplify the modeling of complex semantic ontologies and topological algorithms, still requires more investigation in the future.

Endnotes

1. Protégé <http://protege.stanford.edu/>
2. RDFLib <https://github.com/RDFLib>
3. FuXi <https://code.google.com/p/fuxi/>  
5. GhPython <http://www.food4rhino.com/project/ghpython?utf>
Acknowledgements

The Ministry of Science and Technology of Taiwan supports this paper under the grant number MOST 103-2221-E-165-001-MY2.

References


