Automation in Construction 96 (2018) 224–235

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

From the generation of layouts to the production of construction documents: An application in the customization of apartment plans

Pedro Veloso[a], Gabriela Celani[b,*], Rodrigo Scheeren[c]

[a] Carnegie Mellon University, United States of America
[b] University of Campinas, Brazil
[c] University of São Paulo, Brazil

ARTICLE INFO

Keywords:
House plan customization
Generative design
Shape grammars
BIM
Interoperability

ABSTRACT

This paper describes a design customization system that integrates two aspects of Computer-Aided Architectural Design (CAAD) that are usually developed in separate workflows: the algorithmic generation of designs and the detailed representation of the building. The system's workflow starts with the definition of shape grammar rules by an architect. The rules are then automatically imported into a user interface that allows future owners to interactively custom-design their apartment plans. Finally, the plans are automatically converted into detailed Building Information Models (BIM), which allow the architect to add custom finishes, estimate building costs, and automatically generate construction drawings. We conclude that our workflow could contribute to the real customization of houses and other simple architectural programmes, assuring the quality of the outcomes through shape grammars rules and at the same time reducing the cost of production drawings through automation. The paper ends with some suggestions of improvements in BIM software that would allow its integration with shape grammars and the implementation of our workflow in a simpler way.

1. Introduction

Historically, developments in Computer-Aided Architectural Design (CAAD) have mainly targeted two major goals: the generation of designs and their representation. The first goal includes tools for analyzing local conditions and automatically generating design alternatives in order to create better designs [1], while the second is devoted to creating ever more sophisticated and complete descriptions of designs, increasing the productivity of the architectural office [2].

These two goals have been associated, respectively, with two parallel lines of research. On the one hand, computational design research encompasses algorithmic, parametric and generative design, usually applied to the early phases of the design process. On the other hand, Building Information Modeling (BIM) aims at the rationalization of the representation by means of a single model with the automatic generation of its sub-products, such as 2D drawings and materials schedules. Specific software support each of these research approaches, such as Generative Components and Grasshopper on one side, and Archicad and Revit on the other. Both types of software have been successfully combined in many complex projects, those by Zaha Hadid's and Gehry and Partners' offices being perhaps the most noteworthy.

Some authors affirm that combining parametric design and BIM offers many advantages, such as pushing to the end of the process the decision about the final form from among a large variety of possible solutions [3,4]. However, the workflow integration between them is still the subject of discussion and research, since it has not always been easy or obvious, oftentimes requiring redundant work, due to difficulties in interoperability [4–7]. In the specific case considered in this paper, namely, of integrating shape grammar implementations, not much has been done so far.

In the present paper we describe a workflow that combines shape grammars and BIM modeling, automating both the generation of plans and the production of detailed 3D models and 2D drawings. The project started in response to a real-estate developer's request, who wanted to enable users to interactively customize their apartment's plans. At the same time, he wanted to have a certain level of control in order to assure the quality of the resulting designs. The control over the quality of designs was achieved with the use of the shape grammar paradigm [8], while the technical and economic feasibility of the entire process was achieved through the definition of a workflow in which the conversion between different types of representations was automated. The development of this project was an opportunity to reflect on the importance of integrating the generative design and the Building Information Modeling paradigms in a seamless workflow.

* Corresponding author.
E-mail address: celani@fec.unicamp.br (G. Celani).

https://doi.org/10.1016/j.autcon.2018.09.013
Received 14 May 2018; Received in revised form 17 September 2018; Accepted 19 September 2018

2. Literature review

In order to automate the different phases of this project we used different software and programming languages, and we had to be very careful when transferring data throughout this workflow in order to avoid loss of information. The development of tools that support architects in creating their own design process is one of the big challenges of the CAAD industry nowadays, rather than forcing them to adapt their processes to rigid tool requirements [9].

Janssen et al. [10] describes two different approaches in workflow integration between conceptual modeling tools and BIM systems. The first is the tightly coupled approach, in which the "graph-based systems communicate via the (Application Programming Interface) API of the BIM system, directly instantiating geometry in the BIM model each time the graph-based model is executed". The second is the loosely coupled approach, in which "the graph-based system typically generates an IFC model that can be directly imported into the BIM system". Each one of them can use plugins to facilitate the data translation. The downside of the first is that it only works with specific BIM tools, and of the second is the high complexity in creating customized materialization procedures and loss of quality of the model data.

Toth et al. [9] examined some features to develop a framework for integration, such as user-friendly representations, applying different control flow mechanisms to ensure maximum flexibility, and guaranteeing a repository for system implementation. According to them, "workflow design is likely to be an incremental process in which a number of nodes are combined into a partial workflow, tested by the designer, then further developed and extended" ([9] p. 496).

In fact, one of the reasons why generative design strategies have been mostly restricted to academic or special applications[1] might be the difficulty in integrating them in the usual architectural office workflow. Many authors have addressed this dichotomy.

Computational design has been considered as a paradigm that can enhance creativity [11,12], while BIM has often been considered to limit it [13]. On the other hand, some authors have criticized the ever more complex forms that result from generative software, asserting that they can lead to designs that are too expensive or impossible to build [14], because they lack the concreteness of a BIM model. One common complaint, however, is the lack of interoperability between software that supports each paradigm. Tommasi and Achille [7] point out the challenge in integrating geometry coming from different modeling paradigms[2].

More recently, however, BIM software has tended to incorporate generative capacities, and nowadays most packages include built-in graphic programming environments or communication with them through plugins (Fig. 1). The combination of both types of tools in a single environment brings about the opportunity to challenge the traditional top-down design process, which starts with massing and ends with detailed drawings. With the development of both general and particular aspects of the design in parallel, as asserted by Zarzycki [15]: "...there is an opportunity to establish the interoperability of data, or a bidirectional design process with designers simultaneously working on the general and the specific, within all phases and scales of the project". According to him, this is a sign that the industry is responding to the need for integrating conceptual design and object-based modeling with BIM tools.

The integration between parametric and object-based models can be performed by means of two types of procedures: (1) exporting the geometric model and information protocols; (2) integrating data with the use of plugins. In the first case, a database is generated in a file that requires a manual transfer between the platforms. In the second, the data are transferred via plugins with API connections, which instantly translate the model. In the first case, the Industry Foundation Classes (IFC) open file format is used. IFC (ISO 16739:2013) is "an information-rich object-based file format for representing building information" ([10] p. 584) that was created by the International Alliance for Interoperability (IAI) to facilitate interoperability in the Architecture, Engineering and Construction (AEC) industry. In the second case, the geometric model is imported directly into a BIM tool, and the geometric entities are converted into BIM components, through an Application Programming Connection. In this case, "graph-based systems communicate via the API of the BIM system, directly instantiating geometry in the BIM model each time the graph-based model is executed" [10]. Fig. 1 shows some examples of direct and indirect (via plugin) data integration between different generative and BIM packages.

One typical example in which this integration is needed is the mass-customization of housing plans by users, which requires an interactive generative system, such as a shape grammar, and the automated production of construction documents.

There are many examples of the application of computational concepts for residential plan generation. For example, Kowaltowski et al. [16] implemented a parametric tool that adapts schematic plans to the user's plot dimensions for self-built houses. Duarte [17] developed a shape grammar based on Alvaro Siza's Malagueira houses which also implemented computationally at a schematic level for research purposes. Eloi [18] and Griz et al. [19] developed grammars for the adaptation of existing home layouts to meet the dwellers' needs, also at a schematic level. However, true customization of collective housing seems to be still non-existent, and one possible reason is the difficulty in integrating conceptual studies with detailed design and documentation, which is usually done in BIM software nowadays.

3. Method

The system developed in the present research comprises three parts: (1) the definition of spatial subdivision and union rules by the project architect, based on the shape grammar paradigm; (2) an interactive layout design interface to allow non-designer users to develop custom plans; and (3) the translation of the plans developed into a BIM representation, a step that has been termed "materialization process" by Janssen et al. [10].

3.1. Definition of rules

The method used for the generation of layouts is based on the shape grammar paradigm[3], although the definition of rules did not follow the classical IF/THEN pattern, with the left-hand side of the rule being the shape to which the rule will be applied and the right-hand side of the rule the result of its application. Instead, for practical reasons, we developed a method through which an architect with no programming background can easily define spatial subdivision or union rules in a CAD software, such as AutoCAD or Rhinoceros. The architect simply draws a number of possible transformations in the design process for each shape being altered. Next, the architect draws lines (in a special layer and in the right direction, towards the new shapes) connecting each transformation step. Whenever a subdivision rule can be applied, the lines connect a single shape to two or more shapes. Whenever a union rule is applied to two separate spaces, the transformation then converges to a single state. Fig. 2 shows an example of some possible subdivision and union rules and compares it to the classical shape grammar rule representation. The resulting graph is neither a shape

[1] Some experimental or larger offices have special groups developing this type of integrated generative-BIM applications, such as Gehry Technologies, Zaha Hadid Architects, etc.

[2] These authors actually consider generative and BIM models as subsets of the parametric category, and they distinguish them from the "traditional" or directly modeled geometry.

[3] Strictly speaking, shape grammars only have addition and subtraction rules. Here we refer to our rules as "division" and "union" to better explain what is happening to the rooms.