Xp-GEN: A randomized design tool for non-deterministic digital design methods in architecture and visual design

Burak Pak, Ozan Önder Özener, Assoc. Prof. Arzu Erdem (PhD.)

ITU Institute of Informatics, Graduate Program in Architectural Design Computing
http://virtuvius.itu.edu.tr

Abstract. "Experimental generator" (XpGEN) is a plug-in that allows user to interact with computer for experimental, intuitive and inspirational assistance during the beginning of the architectural and basic design phase by randomly generating multiple design alternatives according to the limitations of the user. The tool is also an experiment to question the physical limits of architectural design.

Keywords. Generative Design Tools, Virtual Architecture, Digital Tools in Architecture

Introduction

For designers, visual speculations assist inventions and visual tokens most effectively represent ideas. They need sufficiently intuitive tools that allow them to use their full spectrum of their abilities and imagination. (Mc Cullough 1996) Digitalization of the design tools and design process has currently shift the conceptual thinking both in real (physical) and virtual architecture. It is also clear that the new paradigm of emerging design process is bound to electronic and digital technologies. The other fact in digitalized architecture is the base with an extremely fluid and (n) dimensioned application space. Thus, we need new and capable tools to practice in this new environment.

Besides supporting the designer and forming the starting points for conceptual design in digital environment is a new approach in digital and virtual architecture. Computer supported generative systems can provide useful inputs for developing different types of architectural products.

Early research about the understanding of the design process led the idea of generative design systems. (Rowe 1987) Researchers such as Ömer Akin, William J. Mitchell and Charles Eastman consider the design process as a problem-solving process based on information processing theory of Newell and Simon. According to Akin (1986), design is a staged and iterative problem-solving process. SEED project which combines LOOS/ABLOOS (Coyne 1991) and GENESIS (Heisserman 1994) is an example of a problem solving software environment that supports the early phases in the design of buildings.

Shape grammars are also a similar computational approach to the generation and analysis of designs. George Stiny (1975) developed the basic landscape for shape operations and relations, including subshape, and their algorithms. Ulrich Flemming (1987) developed a Prolog program that implemented his grammar for Queen Anne houses. Krishnamurti and Earl (1992) developed computational models for three-dimensional grammars. Mark Tapia's (1999) GEdit and Cagan's "coffee maker grammar" are general two-dimensional shape grammar interpreters available on the web.

In most all of the foregoing work, computational problems related to encoding rules and their execution in a computer program took precedence over user interfaces. Thus, these systems were not suitable, or ready, for general use by non-programmers, novice users of shape grammars, or design practitioners. (Knight 1998) Most work on shape grammar implementations has involved straight lines or planar surfaces.(Gips 1999)

The Model Structure

Experimental generator (XpGEN) is a plug-in that allows user to interact with computer for experimental, intuitive and inspirational assistance during the beginning of the architectural and basic design phase by randomly generating multiple design alternatives according to the limitations of the user. These alternatives are 3D sculptors rather than rule based functional layouts. The tool is actually a 3D visual speculator and does not mimic traditional design techniques. It doesn't automate the design process. User's decision is the main concept. The aims are :

• To explore the potentials of digital tools for creating architectural input.
• To implement digital tool to support design studio.
• To enhance architectural projects with still and animated inputs.
• To observe how students use basic digital tools in design studio.
• To take feedback to improve the interface and code structure.

In this model; the starting object and transformation rules are not restricted and can be determined by the user. Evaluation of the generated products will be made by the user(s) with connected modules of XpGEN. The model consists of three main modules,

1. Generator
2. Design file manager
3. XpDAT database

Generator module executes the user defined rules to the starting object. In this version modification functions are controlled by the production engine, the engine allows to use modifiers with different combinations. With these modifiers and user defined parameters the user can try different types of transformations and randomized modifications. The model supports both deterministic and randomized generation strategy.

In random generation process, first phase is to define object and space relationship with the limits of x, y and z coordinates. These settings are defined by maximum limits of desired design space. In the modification menu there are 4 movable modifications; Scale, Twist, Bend and Skew. The other parameters can be listed as;

• Coordinate parameters
• Rotation parameters
• Number of copies
• Name of the starting object

These known and simple modifications are selected to maximize the eligibility and usability of the tool in the design development process. These are the basic object modification rules can be found in most of digital design tools.

In the random creation phase; starting objects are transformed with these modifiers using random variables generated in the script. Also objects are multiplied and positioned in the borders of design space or a defined route. It is also aimed to produce non-platonic shapes for devel-

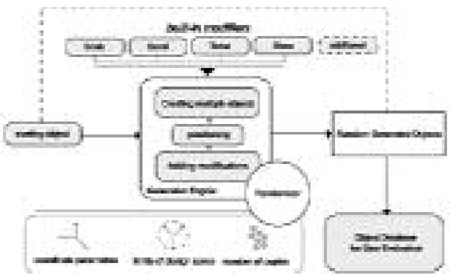

Figure 1: XpGEN Flow Diagram

oping designs. The flow diagram is given in figure1.

XpGEN also supports a deterministic design process. All the parameters defined above can be pre-determined by the user. This feature allows a tightly controlled generation process.

In generative design systems, designers can be overwhelmed by the amount of information they are able to generate in a short time. This process is as though designers are exploring a forest in the dark with a flashlight. (Chien 98) File management and storage modules in XpGEN are simple solutions overcome this problem.

Generated objects can be stored and loaded using the design file manager module. This module is a standalone script which generates a thumbnail of the alternative, scans the directory and finds all alternative file names with the same base name that already exists, takes their count, adds 1 to it and saves a new file and thumbnail with an incremental number.

XpDAT database is formed using a simple script that collects all files the directories and classifies them. XpDAT provides a reporting feature that allows users to compare and evaluate different design alternatives altogether. Another

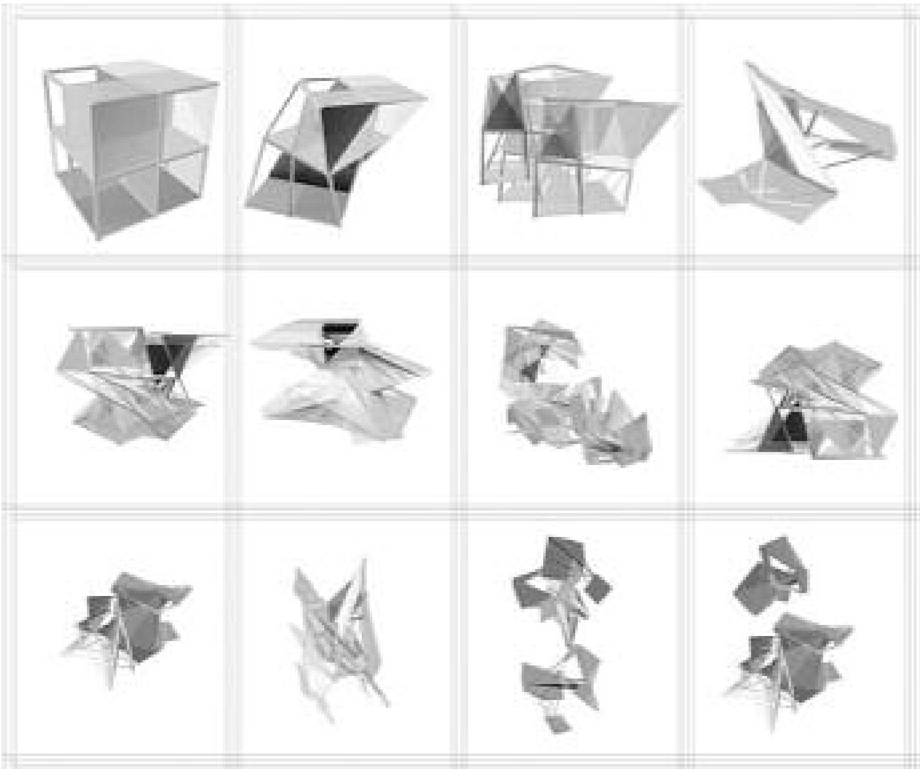

Figure 2: Example of generation process

function of the module is to manage huge number of alternatives and media generated by the user.

Implementation and Future Recommendations

The tool was used in various studios and workshops in the ITU Faculty of Architecture and professional multimedia and graphic projects in ITU New Media Laboratory as well. In these experiments, it has been observed that; the tool supported the object and the structure creation process in digital environment also gave opportunity to try and evaluate variety of design alternatives. It is also observed that tool has increased the students' capacity to visualize abstract 3D symbolic structures as well as modification of digital objects. Consequently, complexity of digital 3D structures became a clear entity and easy to understood.

Animating the generation and including "time" as a parameter of the design process made the process more attractive and more effective. The plug-in has motivated them to develop interesting concepts, expressing themselves and combining different types of media.

In the future the plug-in will be improved by additional modifiers and transformers to create a huge variety of alternatives. The potential use of the XpGEN has a wide area from design education to professional practice both in digital and virtual manner. This tool can be seen as initial example of implementing basic design education to digital medium and connecting design practice with design education.

Acknowledgements

The authors wish to express their gratitude to the participants and students. Burak Pak (M.Sc) and Ozan Önder Özener (M.Sc) are currently research assistants at the ITU Institute of Informatics, Department of Information Technologies and Computer Science. Arzu Erdem (PhD.) is Associate Professor of Design at the ITU Faculty of Architecture,Graduate Program in Design Computing

References

Rowe, P. G.: 1987, Design Thinking, Cambridge, MA: The MIT Press.

Gips, J.: 1999, Computer implementation of shape grammars, invited paper, NSF/MIT Workshop on Shape Computation.

Akin, O.: 1986, Psychology of Architectural Design, Pion Limited

Stiny G.: 1975, Pictorial and Formal Aspects of Shape and Shape Grammars, Basel: Birkhauser

McCullough M.: 1998, Abstracting the Craft: The Practiced Digital Hand, The MIT Press, Cambridge

Chien, S.: 1998, Supporting Information Navigation in Generative Design Systems, PhD. Thesis, Carnegie Mellon University

Flemming U.: 1987, More than the sum of parts: the grammar of Queen Anne Houses, Environment and Planning B: Planning and Design 14, pp. 323–350.

Heisserman J.: 1994, Generative geometric design, IEEE Computer Graphics and Applications 14, pp.37–45.

Krishnamurti R.: 1992, The maximal representation of a shape, Environment and Planning B: Planning and Design 19, pp. 267–288.

Knight T W, 1998, Shape grammars, Environment and Planning B: Planning and Design, Anniversary Issue, pp. 86–91.