WP-BIM: Web-based Parametric BIM Towards Online Collaborative Design and Optimization

Wei Yan¹
¹Texas A&M University
¹wyan@tamu.edu

We present initial experiments of Web-based Parametric Building Information Modeling (WP-BIM) towards collaborative design, modeling, simulation, and optimization. A new framework that integrates Web-based information technology (WebGL graphics, networking, and Web browsers), and design computing technology (visual programming) into parametric BIM is prototyped for the experiments. The integration of Web technology is going to enable online collaborative and user participatory design. Connected through the Web platform, a BIM model, visual programming-based user interfaces for parametric changes, and an optimization algorithm, which may reside in different servers or local computers in different geographical locations, have the potential to be integrated and working together to resolve design optimization problems, especially if combined with cloud-based performance simulation tools. After future development, this may allow architects, engineers, clients, etc. to collaboratively work on a project with up-to-date building data and different design and simulation tools.

Keywords: Web-based, Parametric Modeling, BIM, Collaborative Design, Optimization

INTRODUCTION

In this paper, we present initial experiments of Web-based Parametric Building Information Modeling (WP-BIM) towards collaborative design, modeling, simulation, and optimization. A new prototypical framework is created based on Building Information Modeling (BIM), parametric modeling, and Web technologies. The framework integrates Web-based information technology (WebGL graphics, networking, and Web browsers), and design computing technology (visual programming) into parametric BIM. The important addition of Web technology is going to enable designers’ collaboration online and allow user participatory design. Visual programming uses a graph structure of data flow for creating computer programs to conduct parametric design, simulation, and optimization. Visual programming is being used more and more widely in contemporary architectural design projects. Connected through the Web platform, a BIM model, visual programming-based user interfaces for parametric changes, and an optimization algorithm, which may reside in different servers or local computers in different geographical locations, have the potential to be integrated
and working together to resolve design optimization problems, especially if combined with cloud-based performance simulation tools. After future development, this may allow architects, engineers, clients, etc. to collaboratively work on a project with up-to-date building data and different design and simulation tools of the users’ choices. The integration of these different computing technologies also provides a case study of solving the challenging software interoperability problems.

**BACKGROUNDs**

BIM is a digital representation of a process to facilitate the exchange and interoperability of building information (Eastman et al. 2011) and a digital representation of a product of physical and functional characteristics of a building (NIBS 2008). Semantically rich and object-based, BIM facilitates the creation and management of comprehensive building data, including objects and their properties used in design, simulation, cost estimation, construction, and operation, increasing the AEC process efficiency. In addition, BIM’s parametric modeling capability enables quick, interactive, and real-time design changes (Lee et al. 2006). The relationship between BIM and parametric modeling lies in that BIM contains building objects and their relationships, which can be used to express design intent, while the parametric modeling method helps establish and manage these relationships (Yan 2014). In a BIM project, objects are defined by built-in or user-specified parameters, and external data such as physical and functional data accessed through databases or entered in Graphical User Interfaces (GUI). Parametric modeling enables parameters to be processed by mathematical formulas and computational algorithms before being passed among objects. The formulas and algorithms can be designed based on research and creative design thinking. Integrated together, parametric BIM becomes a powerful design tool for architects.

As a contemporary architectural design method, parametric modeling is important in the design process for creating parametric design models that can generate multiple design options, whose performance (such as energy, daylighting, and functions) can be simulated and optimized. Existing systems, such as Flux (https://flux.io), are able to translate model data among BIM, databases (e.g. Excel), and geometry modeling tools (Rhino/Grasshopper or Revit/Dynamo) and display the models on the Web. However, establishing parametric relationships or constraints of design objects using different modeling tools working with the Web has not been experimented, but is necessary for multidisciplinary collaboration and software interoperability. Therefore, one of the major research aims of this study is the creation of parametric relationships between building objects in the Web-based building information models with various design tools. Unlike existing systems that attempt to display isolated parameters and their static values online, this proposed project is going to establish parametric relationships in the building models that may be displayed and changed online in the design process by designers, engineers, and building users. The purpose of establishing the parametric relationships in the building models is to maintain design intents for parametric studies and design optimization.

**METHODOLOGY**

The methodology used in this research consists of prototyping and experiments with case studies. Previously, a plugin (Autodesk Design Review) of Internet Explorer was used to allow BIM model geometry and database to be viewed online for a case study (Jeong and Yan 2013). The requirement of a plugin in a specific Web browser limited the application and development of the system significantly. This present research has created a new integrated system that utilizes Autodesk Forge Viewer, which can work on most of the modern Web browsers without any plugin (Figure 1), allowing BIM models and their parameters to be displayed in a Web browser directly, and further enabling the interaction between visual programming tools and BIM online. Using Autodesk Forge Viewer API and Model Derivative API, the BIM
model (made in Revit) was uploaded to Autodesk Forge cloud, translated into the SVF format, and displayed in Forge Viewer. BIM 3D models, 2D drawings, and parameters can be displayed in a Web browser using Forge Viewer (Figure 1 left). A user can interactively examine the design model (Figure 1 right). All major current Web-browsers support the model display due to the use of JavaScript-based WebGL technology by the Forge Viewer. This system supports our new research and development to enable parametric relationships between building objects for the Web-based models.

We investigated the use of Flux to facilitate the integration of major visual programming tools for design, including Dynamo and Grasshopper, with Forge Viewer, and developed a prototype for demonstrating the Web-based parametric modeling method using visual programming and Forge Viewer. The prototype is created using JavaScript and Forge Viewer API/Extension. A sample of running the prototype for the communication of Dynamo and Flux is shown in Figure 2. In Figure 3, Flux is working on the background and help the communication between Dynamo and Forge Viewer, through our developed JavaScript program. When the slider of the Translate parameter is changed in Dynamo, the translation data is sent to Flux, and Viewer receives the data from Flux and makes the translation of the building roof in real time. Similarly, the slider of Scale is used to control the scaling of the roof.

After the integration between Viewer and Dynamo through Flux, it’s similar to integrate Grasshopper with Viewer through Flux (Figure 4). Compared to current developments combining Web-based 3D models and simple graph-based programming tools for editing the models, such as Autodesk Project Fractal (https://home.fractal.live) and Flux Flow (https://flux.io), the present prototype utilizes full-fledged visual programming tools: Dynamo and Grasshopper, which support complex parametric, generative design through their comprehensive built-in function and algorithm libraries and a large number of third-party plugins.

Based on the integration of Viewer, Dynamo, Grasshopper, and Flux, we developed a simulation of two users working on the same building model in Viewers: (1) User 1 controls the Dynamo program for the roof translation and scaling with Viewer on Google Chrome; and (2) User 2 controls the Grasshopper program for the roof rotation with...
Viewer on Microsoft Edge. The actual use of two computers to test the collaborative parametric modeling between two (simulated) users was conducted (Figure 5). In the test, one computer runs Viewer and Dynamo to change the translation and scaling, and the other computer runs Viewer and Grasshopper to change the rotation. The transformations are synchronized on the two computers.

**EXPERIMENTS WITH WP-BIM IN DESIGN OPTIMIZATION**

In the present research, we have experimented Web-based parametric modeling supporting design optimization for a simple example. The experiment utilizes a Genetic Algorithm tool (Galapagos) in a visual programming environment (Grasshopper) to automatically generate design options and improve the options for sample design objectives. During the optimization process, Web-based Viewer is able to display the varying design options. A scenario of two designers using different visual programming tools - Dynamo and Grasshopper - to experiment with a preliminary design optimization through the Web is simulated.

In the example as shown in Figure 6, the design objectives are: through changing the rotation and scale of the roof, make the roof:

1. cover the whole building floorplan (50ft x 25ft)
2. cover the front area of the door (i.e. a location point that is 2.5ft in front of the door needs to be covered by the roof.)
3. have minimal area (i.e. the scale factor of the roof should be as small as possible).

While the Dynamo program can still be used to change the translation and scaling of the roof by a user manually (Figure 6 upper left), the Grasshopper program now contains a Galapagos (Genetic Algorithm) node, which uses the rotation angle and the scaling factor of the roof as parameters, and a fitness function that combines the above design objectives for optimization (Figure 6 lower left).

Running the Genetic Algorithm, multiple generations of design options are created automatically and evaluated with the fitness function. In the process, Galapagos inside Grasshopper displays the running of the Genetic Algorithm (Figure 7 lower left), and Viewer displays the multiple design options automatically (Figure 7 right). Figure 7 (right) also displays the resulting optimal design solution after running the process for 41 generations. From the bot-
Figure 5
Two computers representing two users collaborating on parametric modeling.

tom perspective, Viewer shows the resulting rotation angle and scale of the roof. The building is covered by the roof, the door’s front area is also covered by the roof, and the roof area (scale) is the smallest among all the solutions found in the specific optimization run of 41 generations. For parameter, constraint, or objective changes, re-running the optimization process can produce new design solutions. For example, the door front point is set 5 ft farther away from the door (the distance is 7.5 ft now), the results of the new optimization are shown in Figure 8, which also displays the Grasshopper-generated geometry model in the modeling tool Rhino. Note that the building model was originally created as building information model in Revit, and uploaded to Autodesk Forge cloud, translated into the SVF format, and displayed in Forge Viewer in a Web browser.

The changing of the building’s design options can be seen on the Web in real time by multiple users, who will be informed about the optimization process and may eventually contribute to the optimization process from different perspectives.

DISCUSSIONS
The case study was an example of a simplified single objective optimization problem (combining the multiple objectives into a single objective through weighted sum). Architectural design can be regarded as a multi-objective optimization process, where multiple design objectives need to be satisfied or optimized. For example, a project will optimize space layouts, building structure, energy consumption, daylight performance, costs, etc. Each of these building performances can be formulated into a separate design optimization objective, and all together, comprehensive optimal design solutions are sought in the design process. For a complex design project, the search space (the set of all possible solutions) can be very large, due to the large number of variable parameters and their constraints in multi-disciplines. Dividing the search space into subspaces and using concurrent subspace optimization algorithms are good strategies for effective optimization (Lu et al. 2014). However, the studies of concurrent subspace optimization on how to define the subspaces, parameters and constraints inside each subspace, and common or shared parameters and constraints between subspaces, and how to use efficient optimization algorithms, have been conducted mostly in engineering design, such as aerospace, automobile, etc., but not in architectural design. Further development of the WP-BIM system may help provide such a tool for the study of concurrent subspace optimization to facilitate architectural design collaboration.
Figure 6
Sample Web-based parametric modeling and design optimization.

Figure 7
The optimization workflow and results.
CONCLUSIONS AND FUTURE WORK

The prototype integrating Forge Viewer, visual programming tools Dynamo and Grasshopper, the data interoperability platform Flux, and Genetic Algorithm, was developed using JavaScript programming with Viewer API and Extension as well as other mentioned tools. The prototype can run successfully demonstrating the capability of Web-based collaborative parametric design and optimization with visual programming and BIM models. The source code of the prototype is hosted on the online software repository GitHub: (https://github.com/wyanTAMU/coparametricdesign).

Future work includes: (1) extending the Viewer’s model editing functions to enable more comprehensive parametric modeling capability; (2) testing the applications of Web-based collaborative parametric design and optimization with more complex case studies; (3) investigating distributed and concurrent subspace optimization processes utilizing an enhanced framework. Future WP-BIM system will also include different optimization algorithms for solving different problems within subspaces of optimization, e.g. classical calculus for solving optimization problems that can be formulated as continuous and differentiable equations, linear programming for problems that can be modeled by linear relationships between variables, and dynamic programming for those with discrete and stochastic variables. In addition, different Genetic Algorithm tools, including BIM- and visual programming-based Optimo (Rahmani Asl et al. 2014, 2015a, and 2015b) can be included in the framework. It is expected that the Web-based, collaborative parametric design method utilizing different modeling and visual programming tools will enhance the multi-disciplinary and multi-objective design optimization process for architecture.

Figure 8
New optimization results after the door front point is changed to 7.5 ft away from the door.
ACKNOWLEDGEMENTS
This project is funded by the Autodesk Forge Research Grant 2016. The support from the Autodesk Forge team: Ms. Natalia Polikarpova and Mr. Jaime Rosales Duque, from Autodesk, Inc.: Dr. Mohammad Rahmani Asl (project consultant) and Dr. Mikako Harada, and from Flux Data Inc. is greatly appreciated. The BIM model of the historic Elm Springs Plantation mansion (Figure 1) was created by Ms. Leslie Leffke in ARCH652, Fall 2008, at Texas A&M University.

REFERENCES
National Institute of Building Sciences, NIBS 2008, National Building Information Model Standard,
https://www.nationalbimstandard.org