WHAT DO DESIGN DATA SAY ABOUT YOUR MODEL?

A Case Study on Reliability and Validity

HALIL ERHAN[1], AHMED M. ABUZURAIQ[2], MARYAM ZAREI[3], OSAMA ALSALMAN[4], ROBERT WOODBURY[5] and JOHN DILL[6]
[1,2,3,4,5,6] *Simon Fraser University*
[1,2,3,4,5,6] *{herhan|aabuzura|mzarei|oalsalma|rw|dill}@sfu.ca*

Abstract. Parametric modeling systems are widely used in architectural design. Their use for designing complex built environments raises important practical challenges when composed by multiple people with diverse interests and using mostly unverified computational modules. Through a case study, we investigate possible concerns identifiable from a real-world collaborative design setting and how such concerns can be revealed through interactive data visualizations of parametric models. We then present our approach for resolving these concerns using a design analytic workflow for examine their reliability and validity. We summarize the lessons learnt from the case study, such as the importance of an abundance of test cases, reproducible design instances, accessing and interacting with data during all phases of design, and seeking high cohesion and decoupling between design geometry and evaluation components. We suggest a systematic integration of design modeling and analytics for enhancing a reliable design decision-making.

Keywords. Model Reliability; Model Validity; Parametric Modeling; Design Analytics; Design Visualization.

1. Introduction

Parametric modeling tools adapting dataflow graphs have been widely used in many disciplines such as design, engineering, economics, natural sciences, etc. (Johnston et al., 2004; Sousa, 2012). They enable change propagation in and across models, exploration of variations, and multi-level interaction (Aish and Woodbury, 2005). Parametric models can be studied as computer programs and the activities to build them are akin to composing a program (Davis et al., 2011; Ko et al., 2011). This includes most aspects of parametric modeling from inserting a node and writing scripts to using APIs. It is essential that these models are reliable and perform as intended, and, like computer programs, they must be tested for their reliability and validity with a caveat that the testing methods should be unique for design modeling. Here, we use reliability as a criterion for determining if a given model performs without errors under changing conditions; and, for validity, if it produces expected designs with expected performance measures (Lyu, 1996). As

RE: Anthropocene, Proceedings of the 25th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2020, Paper 272 (Preprint).

design information captured and used in design grows rapidly, it becomes almost impossible for the existing computational design ecosystem to assist designers in tracing their decisions in their design models. There is an important need to support traceability, reliability, and validity of complex design models in a collaborative environment (Marchenko et al., 2011).

A parametric model's reliability and validity can be affected by, first, the complexity of its intricate and changing structure (Erhan et al., 2010; Davis et al., 2011), and, second, the complexity of the process through which such models are built (Kasik et al., 2005; Ko et al., 2011). Their complexity increases parallel to the increase in their model's fidelity. The process complexity, on the other hand, can result from the involvement of different people in the modeling. These may be easily underestimated and must be considered to avoid undesirable outcomes. We contend that identifying and agreeing to sound practices for controlling parametric model reliability and validity is crucial. We conducted a case study demonstrating how the reliability and validity of parametric models can be tested in a realistic collaborative design case, and how such design models can be improved using the insight gained from the interactive data visualizations. We refer to the combination of computational design and visual data analytics as *design analytics*.

The next section provides a brief background on parametric modeling focusing on performance evaluation following which we describe our approach and the case study on how reliability and validity of such models can be tested and improved. Recent works, e.g. Matejka et al. (2018) and Fuchkina et al. (2018), have explored the use of interactive data visualizations for design space exploration. In the context of evaluating parametric models, the work by Beham et al. (2014) illustrates the use of interactive data visualization and analysis for the purpose of identifying invalid geometry, whereas the work by Nagy et al. (2017) demonstrates the use of heat-map visualizations for assessing the quality of the design space endowed by a parametric model (in terms of its scope and internal structure). In this study, we show how visualizations can facilitate debugging-like scenarios and motivate closer inspection of the parametric models in question. We conclude with recommendations for design teams and software developers to control complexities in parametric design models, and for this, the need for integration of design analytics throughout design workflows.

2. Background

2.1. PARAMETRIC MODELING AS PROGRAMMING

While graph-based parametric models share many similarities with imperative text-based programming, an important distinction arises from the fact that they make visual modeling possible using nodes and links and have an execution model adapting concurrency (Sousa, 2012). In addition, while inputs to a text program are seen as test cases to which the program must correctly react, inputs to a parametric model can be seeds for exploration. Designers, normally, have no obligation to maintain them or verify that they would work reliably with users' inputs as is the case with most other software.

2.2. PERFORMANCE EVALUATION IN DESIGN PHASES

Incorporating the performance metrics in design models from conceptual to the final stages of design has become a common practice (Shea, 2005; Anton and Tăˇnase, 2016). For this, designers use parametric design tools with analysis tools, such as EnergyPlus, Radiance, Daysim, and OpenStudio. This combination allow designers to estimate the design performance as early as possible. However, they are assessment-oriented rather than focusing on dynamically supporting the decision-making (Touloupaki and Theodosiou, 2017). Nembrini et al. (2014) discuss the advantages of using a coding interface both to describe building form and conduct performance simulations. Their approach aims to address performance-related design questions at the early design stages. Architects use parametric modeling combined with various performance analysis software to influence form exploration in the early stages of design that may lead to more informed, efficient and meaningful solutions (Anton and Tanase, 2016).

3. Motivation and Approach

Generated design instances are to be evaluated in the context of a performance-driven design process (Shea et al., 2005; Anton and Tanase, 2016), where the reliability and validity of the models become more pronounced. The performance computation expects specific geometric fidelity and in specific data structures; any violation of this contract will result in unreliable or invalid outcomes. Considering these needs, the creators and users of parametric design should be aware of the importance of building reliable, scalable and reusable models as do their counterparts in the mainstream programming communities. The need for best practices when working with parametric models, especially when performance matters, is therefore a necessity and concepts such as Patterns for Parametric Design (Woodbury, 2010) can be part of the answer.

By generating alternatives, we not only explore the design space, but we simultaneously examine its behavior under various execution scenarios. When multiple designers work in parallel at a form-finding problem, we can expect more diverse forms and form-to-performance evaluations. To study this environment and explore the role of data and its visualization in a design realistic context, the case study we conducted focuses on experimenting with reliability and validity testing and improvement of the collaborative parametric design Setup created during SmartGeometry (2018).

4. Case Study: Collaborative Parametric Design of A High-Rise Building

This case study examines design data from different designers approaching the same design problem. The analysis of the data used design analytics methods. We applied this analysis to a sample of issues arising from the use of parametric modeling systems on complex design problems.

4.1. THE CASE STUDY CONTEXT

The case study is based on our observations of the process and a close study of the collaborative parametric design setup developed in SmartGeometry (2018)

'Inside the Black Box' group. The group comprised 19 participants who were designers, programmers, and visualization developers. The group worked on the design of a mixed-use high-rise building located in a city context with a range of real-world form and performance concerns, such as land-use, functional-spatial distribution, heat loss and gain, solar exposure, view quality, etc. The participants decided to focus on 13 performance criteria in their alternative solutions. One group of participants, who are professional designers, software developers, or parametric modelers, worked on developing the work environment setup, which is composed of performance evaluation, an inputs generator, and server-side data-sharing components (Figure 1). The five designers in the cluster developed six different proposals for the mixed-use high-rise tower design and generated alternatives. We then used their proposed alternatives in our case study.

4.2. DESIGN ANALYTICS WORK ENVIRONMENT SETUP

Our design analytics work environment for testing reliability and validity (Figure 1) is consisting of the following components. First, a parametric model that specifies and generates geometries for alternatives design. The second component receives input parameters and changes input values algorithmically to automate serial generation of alternatives. Parametric design models are combined with a third component which includes parametrically defined context with site, view, geography, and the information regarding other structures around the site. The most complex part of this setup is the component where building performance measures are calculated. This is composed of in-house developed or third-party modules linked to the context and parametric model components. The setup is connected to a database on a dedicated Speckle (2019) server to store and retrieve form and performance data for each alternative, which are further analyzed using Tableau (2019) and our custom-designed set of design analytics tools.

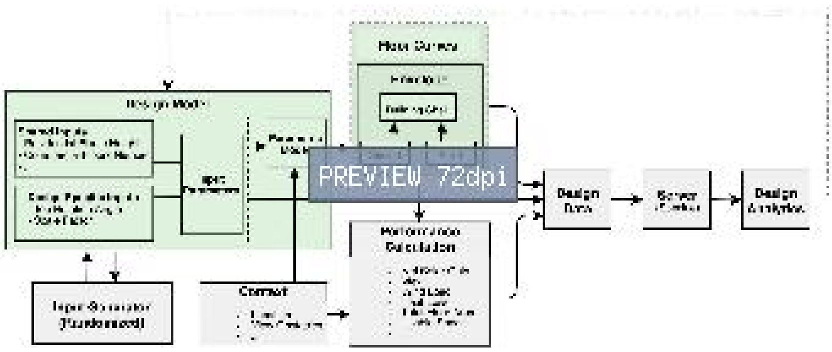

Figure 1. Work Environment Setup for Design Analysis: Design Model, Floor Curves and Envelope (developed by the designers and required for the Performance Calculation module).

We used data from 250 design alternatives generated using parametric models developed by workshop participants. The data associated with each alternative comprised: (a) the input parameters used to generate it; (b) the form and geometry; (c) the 13 different performance values for, e.g. floor area by function, solar gain, cantilever areas, etc. Each alternative was denoted with a unique ID that includes an instance number and a symbol denoting the surname of the model developer.

4.3. CHALLENGES AND PROBLEMS IDENTIFIED

The parametric models in the case study were complex. They presented many issues: they comprised multiple Grasshopper node clusters nested deep within other clusters ; the dependencies between the nodes were difficult to trace; the logic of the dependencies was not obvious, etc. In addition, the modules (e.g. custom-script nodes) in the models were developed by different people that were not directly involved in this particular project. Therefore, the changes in the design models and the changes in the performance calculators might not have been synchronized. The inconsistencies due to versioning in the models increased the threat to reliability and validity of the work environment setup. For example, we identified that there were more than 10 different versions of the generator module created in the workshop. In addition, after creating visualizations of the initial data from 250 designs in Tableau (2019) and with our custom analysis tools, we discovered unexpected patterns on performance calculations. This motivated us to closely study the work environment setup and the models in it. For example, we questioned "what the trends in view quality are" and "how the different types of functions affect heat-loss". These visualizations hinted to possible threats to the validity and reliability of the setup that were not obvious at the outset. Below we discuss some of the salient issues that we observed in the work environment setup.

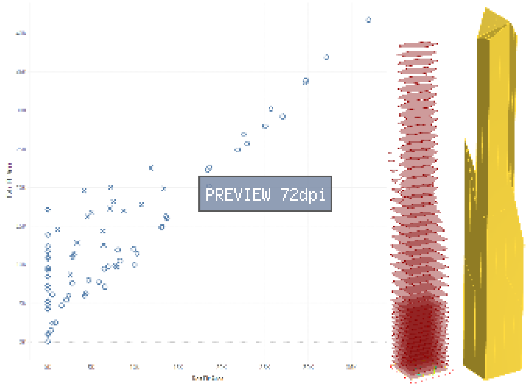

Figure 2. Left: Scatter plot of Total Floor Area vs. Residential Floor Area. Crosses show unexpected outliers. The vertical strip (diamonds) on the vertical axis are alternatives with no residential floors. Right: Floor planes showing segmentation.

Unexpected Correlation. The Total Floor Area (TFA) is the sum of the areas of the residential, commercial and retail floors of the building, hence we expect a correlation between each of them and TFA. Instead, as Figure 2 (Left) shows, we see that some of the alternatives do not follow the expected trend. At first, the reason was not clear. We looked at the parametric model that generated these alternatives. This revealed that when a floor level comprised multiple floor plane segments, e.g. because of gallery openings in the floor, the module calculating TFA considered only the very first segment and ignored the remaining on the same floor level. A further investigation showed that this was due to a mismatch between the tree data structure containing the floor segments and the list data structure expected by the TFA module. This made TFA frequently smaller than the actual sum of the building floor areas. To verify this assumption, we contrasted alternatives close to the trend with outliers, finding that the outlying alternatives tended to have floor planes with more than one segment (Figure 2 (Right)).

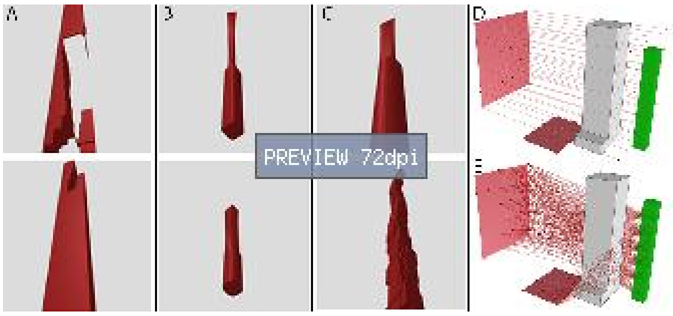

Figure 3. (Left) A: The same parametric model, similar view and height but the top alternative form is malformed. B: The same parametric model, similar view despite very different heights. C: Different parametric models, similar view and height despite differences in the building's shells. (Right) D: Original view calculation. E: Updated view calculation.

View Calculation: In comparing view qualities, we noticed that models with different sizes produced similar view quality values (Figure 3 A, B, and C) in unfamiliar units (degree/m2). Our close study showed the following: (1) In design practices, the view from a building should consider targets in multiple locations and directions. However, the view module was set to one target; (2) The module ignored the obstacles in the context; and, (3) The line of sights were projected from the target surface to the building, as opposed to from different points on the building to the target. In addition, this module used an arbitrary number of parallel projectors that ignored building size. After identifying these three issues in the original view module (Figure 3 D), we developed a new view approach (Figure 3 E) that projects view lines from different points at each floor in the building to multiple view targets and considering the obstacles in the context.

Heat loss. While studying heat loss values in the case study, two experts in our team recognized unexpected values in, which hinted at a possible problem with the calculations. The nested clusters calculating heat loss and heat gain of a building hid the logic and the data visualization showed unexpected results (Figure 4 (Left)), which, in turn, implied that the model might be invalid. Our first assumption was that the relevant module in the performance calculation, Heat Loss, was not working correctly. Therefore, we examined this module to find probable errors. We found the heat loss calculation used equations inconsistent with the standards: they were calculating heat loss due to transmission and ventilation incorrectly. After updating the model with new formulae, we generated a new set of solutions to verify the updates.

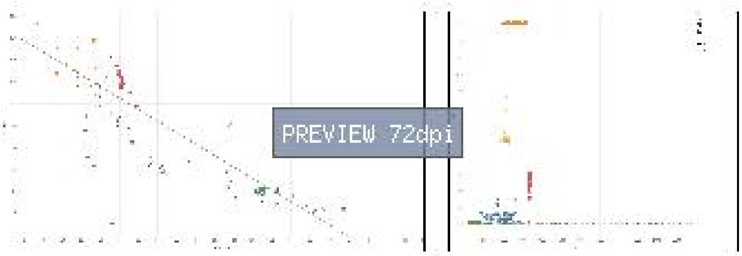

Figure 4. (Left) Plotting energy use vs total floor area, we expected energy use to be positively correlated with the total floor area. (Right) Scatter plot of building height versus cantilever area for 250 alternatives from six different design proposals.

No Variation. Figure 4 (Right) shows a scatter plot of the height versus cantilever area of the generated alternatives. This visualization helped us discover that alternatives generated by the model K do not vary in height, while those generated by model J do not vary in the cantilever area, contrary to what the designers intended. This flagged a problem in both models.

Input Generator. This module generates random values for each input parameter in its set range, from which the model generates a new alternative with the new input values. In the data, we noticed a discrepancy between the generated input values and alternatives. A close inspection showed that the generator module was making assumptions about the time that an alternative would take to be fully computed and prematurely generated new inputs values. However, reliably generating alternatives requires ensuring that the propagation solver finishes computing the current geometry and performance before a new cycle starts.

Table 1. The sample problems identified on the reliability and validity of models and the corresponding lessons derived when addressing them.

#	Lessons	Issues				
		Unexpected Correlation	Views	Heat Loss	No Variation	Input Generator
1	Using Alternatives as Data	x	x		x	
2	The Need for Forms and Visualizations	x	x		x	
3	Matching Input to Actual [illegible] Requirements	x				
4	Using Multiple Parametric Models	x	x			
5	Experts Required (early stage involvement)		x	x		
6	Reproducibility for Revisiting Past Instances	x			x	x

5. Findings and Lessons Learnt

We noted a set of recurring issues and developed a set of solutions or approaches that can be used in addressing the issues in this case study. Table 1 below highlights the mapping between these lessons and the sample issues we discussed above.

Using Alternatives as Data. When we work with a single design alternative at a time, we can only observe the problems that this alternative reveals. Observing a pattern as in Figures 2 and 4 is not possible unless we generate many alternatives and use a data visualization in which the pattern is visible. Data from the alternatives can cover a wide range of test cases demonstrating the model's behavior under a diverse sample of generated input.

Need for Forms and Visualizations. The quantitative performance evaluations and design forms (geometry) present two different aspects of design, complementing each other. Computed design evaluations capture only a fraction of the criteria that may be in action, while geometry at low fidelity can be misleading without further details. For example, the alternatives in Figure 3 (Left) had a similar view performance but the differences between them were only apparent by contrasting their forms. Similarly, the forms in Figure 2 explain the deviations in the Total Floor Area of some buildings.

Matching Input to Performance Module Requirements. Before starting the geometry modeling process, the designer needs to be aware of the requirements and limitations of the performance modules that are suitable for the task. This is so they can provide the right data structure or level of detail. The miscalculation of the Total Floor Area in Section 4.3 shows an example of passing the wrong data structure or so-called data type mismatch.

Using Multiple Parametric Models. For most of the parametric models, the Total Floor Area module worked properly but the expectations were violated for a single model that produced unexpected forms. This highlights the advantage of challenging the module's generality by subjecting it to different design models. Furthermore, when we work with multiple design models, we can learn about the nature of the forms that could achieve high values for given performance criteria. For example, Figure 4 (Right) shows that the forms generated by the model J had better cantilever areas than those of the rest.

Experts Required (early-stage involvement). While it is possible for a correlation pattern or an outlier to raise questions when observed in a data visualization, an expert knowledge is necessary for proper interpretation. Although nothing immediately catches our attention on the plots of heat loss against other performance criteria, an expert with experience in designing buildings in the same context as in the case study, might better identify the expected ranges of heat loss.

Reproducibility for Revisiting Past Instances. In general, designers reproduce an alternative to revisit and reevaluate it against new or updated designs. Achieving this requires retrieving both the input parameters and the model that was used to generate it. In our case study, due to the generator problem discussed in Section 4.3, the reported inputs could no longer be guaranteed to generate the same outputs. Furthermore, because of the dynamic nature of the design process, the parametric model is expected to change in time so that the reproduction is required. It is important then to save the full design model and the correct input values when an alternative is recorded. Re-creation of the design may be needed for, for example, when investigating the source of an error, updating an alternative considering the new information about its performance, or improving form.

6. Conclusion and Discussion

Embedding performance calculation modules in each design module increases a parametric model's complexity and hence affects the models' reliability and validity. For example, each time a performance module changed in the case study, every model had to incorporate the new version, which interrupted the design process. On a practical side, the computing time taken by the performance modules in the model hindered the agility and rapid feedback into the design. Depending on the complexity of the model, the propagation of changes of one parametric model could take between 5 and 30 seconds. We also observed some software-specific issues when performance evaluators developed in a different computer on which the models built. To address these issues, we propose a new collaborative design Setup that decouples the design models from performance evaluation modules (Figure 5). We refactored the setup used in the workshop to separate the concerns and reduce its complexity: while a designer is working on an alternative, he or she can push the model to the server for evaluation. This eliminates embedding performance modules in the model, and hence, reduces complexity. The performance evaluation modules can function in a dedicated model without replication in each design model.

Figure 5. Original (Left) versus proposed (Right) architecture for the collaborative Setup.

The proposed Setup also integrates interactive design data analysis (design analytics tools) into the process. Designers may make better sense of their work when visual analytics is integrated into their workflow. They should be able to study form and geometry details along with performance data derived from design alternatives. The systems must make data available by mapping them on the model, in the model, or on separate visual data analytics interfaces as argued by Erhan et al. (2010) . However, the workflow for creating and use of these data visualizations should not create an extra task layer in design.

The mismatch between the fidelity of the parametric models and the information required by the performance modules could be another source for errors. The performance modules should be flexible enough to handle varying input conditions . It is crucial to access different performance evaluation modules that are suitable for different levels of design abstraction. For example, calculating heat loss for a low-detail design model will require different input and rigor than a highly-detailed design model. The Setup should provide matching requirements of performance modules with the fidelity of the design model.

As demonstrated through the case study, interactive visualization of design data can improve the reliability and validity of parametric modeling in a collaborative design setting. A bottleneck in achieving this is that parametric modeling and visual analytics are often discrete workflows. To alleviate this bottleneck, we need further and more systematic studies for exploring how the integration of these two workflows, i.e., parametric modeling and design analytics, can be further achieved.

Acknowledgment

This research was supported by the MITACS Accelerate Program of Canada [IT12326] in partnership with The Boeing Company and Stantec Vancouver. We thank the SmartGeometry 2018 'Inside the Black Box' cluster champions from Dialog Inc. and the participants for sharing their work with us for this study.

References

"SmartGeometry" : 2018. Available from <https://www.smartgeometry.org>.

Aish, R. and Woodbury, R.: 2005, Multi-level interaction in parametric design, *Smart Graphics*.

Anton, I. and Tănase, D.: 2016, Informed Geometries. Parametric Modelling and Energy Analysis in Early Stages of Design, *Energy Procedia*, **85**, 10-1016.

Davis, D., Burry, J. and Burry, M.: 2011, Understanding visual scripts: Improving collaboration through modular programming, *Int. Journal of Architectural Computing*, **9**(4), 361-375.

Erhan, H., Salmasi, N. and Woodbury, R.: 2010, ViSA: A Parametric Design Modeling Method to Enhance Visual Sensitivity Control and Analysis, *International Journal of Architectural Computing*, **8**, 461-483.

Fuchkina, E., Schneider, S., Bertel, S. and Osintseva, I.: 2018, Design Space Exploration Framework, *Proceedings of the 36th eCAADe Conference*, 367-376.

Johnston, W.M., Hanna, J. and Millar, R.J.: 2004, Advances in dataflow programming languages, *ACM computing surveys (CSUR)*, **36**(1), 1-34.

Kasik, D.J., Buxton, W. and Ferguson, D.R.: 2005, Ten CAD Challenges, *IEEE Comput. Graph. Appl.*, **25**(2), 81-92.

Ko, A., Abraham, R., Beckwith, L., Blackwell, A., Burnett, M., Erwig, M. and Scaffidi, C.: 2011, The state of the art in end-user software engineering, *ACM Comput. Surv*, **21**, 1-21.

Kolari'c, S.E.: 2017, CAMBRIA: Interacting with Multiple CAD Alternatives, *Computer-Aided Architectural Design. Future Trajectories*, Singapore, 81-99.

Lyu, M.i.c.h.a.e.l. R.: 1996, *Handbook of software reliability engineering*, IEEE computer society press CA.

Madkour, Y., Neumann, O. and Erhan, H.: 2009, Programmatic formation: practical applications of parametric design, *Int. Journal of Architectural Computing*, **7**, 587-603.

Marchenko, M., Behrens, B., Wrobel, G., Scheffler, R. and Pleßow, M.: 2011, A New Method of Visualization and Documentation of Parametric Information of 3D CAD Models, *Computer-Aided Design and Applications*, **8**(3), 435-448.

Matejka, J., Glueck, M., Bradner, E., Hashemi, A., Grossman, T. and Fitzmaurice, G.: 2018, Dream Lens: Exploration and Visualization of Large-Scale Generative Design Datasets, *ACM SIGCHI Conference on Human Factors in Computing Systems*, 1-12.

Nagy, D., Villaggi, L., Zhao, D. and Benjamin, D.: 2017, Beyond heuristics: a novel design space model for generative space planning in architecture, *Proceedings of Acadia 2017*.

Nembrini, J., Samberger, S. and Labelle, G.: 2014, Parametric scripting for early design performance simulation, *Energy and Buildings*, **68**, 786-798.

Sousa, T.B.: 2012, Dataflow programming concept, languages and applications, *Doctoral Symposium on Informatics Engineering*.

Touloupaki, E. and Theodosiou, T.: 2017, Performance simulation integrated in parametric 3D modeling as a method for early stage design optimization: A Review, *Energies*, **10**, 10-3390.

Woodbury, R.: 2010, *Elements of parametric design*, Taylor \& Francis Group.