Designing With Data: Moving Beyond The Design Space Catalog

ABSTRACT
Design space catalogs, which present a collection of different options for selection by human designers, have become commonplace in architecture. Increasingly, these catalogs are rapidly generated using parametric models and informed by simulations that describe energy usage, structural efficiency, daylight availability, views, acoustic properties, and other aspects of building performance. However, by conceiving of computational methods as a means for fostering interactive, collaborative, guided, expert-dependent design processes, many opportunities remain to improve upon the originally static archetype of the design space catalog. This paper presents developments in the areas of interaction, automation, simplification, and visualization that seek to improve on the current catalog model while also describing a vision for effective computer-aided, performance-based design processes in the future.

An example design space catalog built from Heinz Isler's famous Infinite Variations of Concrete Shells sketch. Computation can enhance the traditional design process by providing performance feedback and guidance even in conceptual design.
INTRODUCTION

The use of large quantities of data for predictive purposes, whether collected from past actions or generated by first-principle simulations, has the potential to revolutionize many fields. Even in professions that are historically dependent on human intuition, reasoning, and creativity, data can be used to supplement or enhance human activities. Chess and radiation oncology are two examples where the ability to project future outcomes has fundamentally changed how these activities are pursued (Somers 2017). Based on computers that simulate millions of possibilities in great detail, chess players can see breakdowns of their mistakes, train with customized game scenarios that focus on their weaknesses, and if allowed, view real-time information about how each potential move increases or decreases win probabilities. Similarly, doctors can use physics models of radioactive particles to decide precisely where to focus beams of these particles during surgery. Rather than forcing experts to submit to the will of the computer, data merely improves their ability to predict future results and make corresponding adjustments.

As computational methods continue to develop, architects are taking advantage of the many data streams available to them during design. When compared to other fields, however, specific difficulties arise while injecting simulation into the process. The design of a building is much more complex than a game of chess in terms of parameters, stakeholders, and choosing exactly what “winning” means, as each building design is a custom process to some extent. Nevertheless, researchers have produced a wealth of simulation tools that are available to predict how a building behaves, and at least some broad definitions of a design success—energy efficiency, low embodied carbon, high daylight autonomy and thermal comfort, economic and social sustainability—are both universally recognized and increasingly quantifiable. In order to achieve the goal of data-driven design in the form of expert–computer collaboration, these feedback streams and measuring sticks for performance must be intimately linked to the design process itself.

Unfortunately, despite advances in the speed and accuracy of predictive simulations, current technological and organizational barriers often prevent practitioners from fully integrating simulation data, which has traditionally come from engineers or other specialists, into their design workflows. In many cases, building-design projects have already moved into design development or are otherwise frozen in terms of massing or geometry before all relevant performance simulations have been conducted. Even for advanced designers comfortable with data-driven design, the state of the art is essentially to generate a variety of potential design options, view visualizations of their performance, and pick the best option. Functionally, the creation of a design catalog resembles older, more analog procedures for design optioning—computation has sped up the process and provided some notion of relative performance between options, but predictive simulations have not yet reached their full potential to change how we design.

When considering the broader convergence of computation and design, optimization has been offered as a way to move beyond data feedback and push towards guidance, which makes the computer a more active participant in the decision-making process. In mechanical and aerospace engineering, engineering systems, finance, and elsewhere, designers and other professionals have successfully applied optimization to a wide variety of problems. However, most optimization techniques work best when we know exactly what the objectives are and how much they matter compared to one another, and we are willing to accept whatever answer the computer produces. Optimization techniques can certainly assist in improving upon the catalog, but in order to be most useful in architecture, they must be adjusted to account for non-quantifiable objectives, creative flow, and general problem fuzziness.

This paper describes ongoing research that aims to advance the current catalog paradigm and infuse the traditional design process with ways to interact with data and optimization methods more effectively. This research is broken into four categories: interaction, automation, simplification, and visualization. Each area is presented by describing the specific challenge related to design tools and information processing and the responding disruption that has the potential to overcome this challenge. In addition to research contributions that have resulted in tools that are already in development or fully functional, this paper lays out a vision of future pathways that have the potential to revolutionize the architectural design process by enabling effective human–computer cooperation.

BACKGROUND

Most current practices surrounding performance-driven design are based on the concept of the design space. In optimization, the design space is formed by the set of all possible solutions to a problem as determined by the design variables and constraints. Typically in architecture, the design space is made of primarily geometric variables, and different options can be visualized accordingly, but non-geometric properties can also be used as variables. However, each problem can also be described by its objective space, which represents how well a certain design performs. Designs that perform similarly may not look the same, and the reverse can also be true (Figure 2). Thus, the design space should be explored with reference to the objective space. With the advent of parametric modeling (Monedero 2000), it
became relatively easy to create a parametric logic and evaluate the different possibilities that reside within a design space, especially for early, conceptual design. A reasonable next step was to combine visualizations of the design space with a presentation of designs in a "catalog" or "design-by-shopping" (Balling 1999) format to designers for selection (Stump et al. 2003). Along with the gradual advancement of performance simulations, the combination of data visualization and catalog presentation has significantly upgraded the traditional decision-making process in architectural design (Tsigkari et al. 2013).

Yet there is still much room for improvement on the standard model of the catalog. New research can pursue both innovations to the catalog itself, or fundamentally different ways of manipulating geometry in response to data-driven, computational feedback. Future developments must overcome a number of clear barriers, however, towards the goal of effective human–computer collaboration in architectural design. Through a mixture of literature review, new techniques, and short design examples, the next section describes current efforts and identifies remaining opportunities.

**METHODS AND RESULTS**

**Moving Beyond: Interaction**

When pursuing performance-based design, the goal is to implement human–computer interaction in a way that maximizes the synthetic and analytic strengths of both entities. However, developers of such computational environments must delicately balance the adversarial desires for freedom and guidance during exploration. Since the concept of guidance is strongly related to optimization, one of the initial challenges taken on by computational researchers in architecture is the adaption of optimization workflows for more interactive, creative design purposes. When used by architects, standard optimization can push solutions to the edge of the design space, which may not be geometrically or visually appealing, and it has trouble dealing with multiple objectives when the relationships between these objectives are not mathematically defined. Even when optimization is used to create sets of results for review rather than a single design, there is no guarantee that the differences between generated solutions are meaningful enough to be compelling as part of a brainstorming tool.

An initial response to these problems is to relegate the computer to pure feedback and allow the human to perform all prioritization and design synthesis. Brown and Mueller (2016) have experimented with free exploration using live, multidimensional feedback and demonstrated that it can have a positive effect on performance outcomes. However, many designers want clearer guidance. Researchers have developed various forms of interactive optimization, where architectural preferences are expressed progressively during cooperation with an algorithm. Rather than in a priori or a posteriori optimization, where designers numerically define priorities between objectives beforehand or decide on them afterward when viewing a catalog, interactive optimization ensures that expert designers and computers can share input during the process.

A number of tools that employ interactive optimization do so through evolutionary algorithms. In interactive evolutionary algorithms, users can select desirable geometries in one generation of a parametric design and use those designs to breed the next generation. StructureFIT (Mueller and Ochsendorf 2015) implements interactive evolutionary design in a browser-based tool for 2D trusses, and Stormcloud (Danhaive and Mueller 2015) extends this methodology to any parametric design defined in
Grasshopper, for any type of performance evaluation. Schneider et al. (2011) have applied interactive evolution to layout problems, and Harding (2016) has used interactive evolutionary techniques to evolve both the design variables and the topology of the parametric logic itself.

Although evolutionary optimization processes naturally reward and encourage diversity through mutation and crossover, sometimes even these can appear too deterministic or constraining. In response, the authors have developed methods for measuring parametric design space diversity and forcing computers to generate more geometrically interesting and architecturally compelling design spaces. The results of a diversity-driven design process are compared to more traditional optimization workflows in Figure 3, which uses the case study of a tower design. The parametric definition of the high-rise tower contains ten variables related to the floorplate shape, size, and twist of the building, and the volume of the structure is used as a quantitative objective, since it relates to usable floor space. In the first row, an evolutionary solver was applied to the design space to find a catalog of ten geometries that are nearest to the desired volume. The design space was then randomly sampled, and the ten designs closest to the target volume were chosen as possibilities. Finally, designs close to the target volume were attempted in ten different combinations, with the process returning the most diverse catalog generated. Although the entire design space contains more diversity than this final set if performance is totally neglected, this diversity-forcing technique immediately returns a set of designs that closely matches the desired volume and is visually expressive, while at the same time more manageable to choose from than the entire sampled set.

Another challenge to implementing truly interactive, data-driven design processes is the time required to run valuable simulations. As described in Brown et al. (2016), most optimization-based design methodologies can be broken down into two categories—real-time interaction and option generation—and there is a significant difference in the creative process between the two. If the evaluation of performance comes from an analysis of previous outcomes, a quick geometric calculation, or a simulation that runs within a second or two, current parametric modeling tools enable truly immersive, interactive design space exploration. However, for many performance evaluations typical to building design, such as physics-based energy or daylighting models, simulations are too slow to analyze options in real time. This reality has perhaps restrained data-driven design to catalog-based methods more than any other single barrier. Computers will continue to get faster and researchers will develop more efficient calculations, but simulation time remains a significant challenge to implementing real-time interactivity.

Surrogate modeling has the potential to improve the response time of key simulations. This technique, which uses statistical approximation methods rather than live simulations to inform the designer of performance, can be implemented in interactive design environments in lieu of full calculations. Although
Moving Beyond: Automation

Another challenge facing the proliferation of data-driven design tools, which often require parametric or similar algorithmic modeling, is the time and knowledge required to properly formulate and analyze a meaningful design space. Especially when using a directed or semi-directed optimization workflow, this process involves determining which constraints, variables, and bounds are important to the problem. Even as design software becomes more integrated, architectural designers interested in performance-based conceptual design must commit a significant amount of effort to manually creating a model and testing which parts of the design space are most worth exploring.

While using a surrogate model instead of actual simulations, performance-based exploration of form is often possible in real time with only small errors. Although the degree of error depends on specific problems and methods, this approximation of rooftop PV potential closely matches the simulated results.

As approximation techniques become more viable for early-stage exploration, the set of side-by-side, static options of the design catalog can give way to a more dynamic, interactive model with live performance feedback in multiple dimensions. Consider the example of a parametric exploration of rooftop PV potential for a courtyard building, illustrated in Figure 4. As the pitches of the roof are manipulated, the amount of energy that can be produced by a photovoltaic installation on their surfaces also changes. This effect on annual PV potential was modeled using the parametric plugin Archsim (Dogan), assuming a location in Boston and a north-south orientation of the long axis. Each simulation takes approximately 3 seconds on a standard desktop computer. The authors conducted 20 simulations and used the data points to train and validate a surrogate model of the objective function, after which the approximation provided feedback effectively in real time (within a few milliseconds). The approximate results for this problem are fairly accurate, and the faster response time makes the difference between a set of static options versus a truly interactive tool, an effect that is magnified for more complicated designs and corresponding simulations.
Although Davis et al. (2011) have proposed using the principles of modular programming to make model setup easier, the ability to outsource part or all of this process, even in restricted cases, would be a substantial breakthrough in the accessibility and utility of data-driven, performance-based architectural design.

Concerning design space testing, various efforts have been made to devise systems that automate the cataloging of individual designs and index their performance. Most of this software is integrated into typical CAD environments, such as Rhinoceros and Revit. Recent available tools include GenoFORM (Genometri Ltd), Octopus (Vierlinger), ParaGen (von Buelow 2012; Turrin et al. 2011), and Design Space Exploration (Brown et al. 2016). Eastman (2009) similarly focuses on extracting and presenting BIM information in the early stages of design. Although these programs have the potential to greatly improve the speed at which design spaces are sampled and evaluated, they are still significantly dependent on human specialists to construct the parametric logic, including variables, bounds, and how they interact. A more revolutionary technology would be one that can automate design space formulation itself. In related research, Harding and Shepherd (2016) have experimented with computer-generated parametric script definitions. Based on such developments in architecture, one can imagine a performance-based modeling environment in which a designer need only sketch an initial design, apply basic physical properties, and identify key performance objectives, while data-driven techniques automatically generate a parametric possibility space for the provided geometry. This geometric design space would ideally include only variables that have a significant impact on overall performance or architectural expression, and contain only a few global human settings such as model flexibility.

One possible approach to computational problem setup is to begin with an initial geometric sketch and consider possible variable manipulations individually. A first step towards automated parametric generation within this strategy is to test possible variables for a provided geometry and return a ranked list of their relative importance to the performance of the design. Various statistical techniques exist for calculating the controlling variables of a complex model, although some require a robust dataset in the first place, or many performance evaluations to build this dataset. One quick method for calculating variable importance is to use an orthogonal matrix of potential variable settings to sample the design space, and then calculate the average effect of each variable setting on the overall performance of the design. The variable movements that result in the largest perturbation are most important to the problem, and are returned to the designer for exploration.

This technique has been implemented on the example of a trussed roof canopy, illustrated in Figure 5. The shape of the roof canopy is determined by the vertical displacements of control points laid out on a grid on both the top and bottom of a space truss. For this problem, the authors have calculated the average effect of each variable on the performance of the roof, as defined by amount of structural material required, represented by the strength of the colored dots in the image. This method could be generalized for certain architectural typologies so that all potential variables are simulated and tested automatically after...
an initial input sketch. For example, in any truss-like structure the coordinates of each node, as well as intermediate variables that control multiple node locations, could be tested for performance sensitivity, after which the designer would choose the number of variables or threshold of importance he or she desires before the computer automatically generates the parametric model. A computer could also test performance sensitivities for each variable, identify ranges that are important to the design problem, and generate one global slider for model flexibility that automatically sets each variable bound. Once these systems are realized for general architectural problems, such as locating structural elements or building massing, the act of computational, early design space exploration could be extended to those who are largely unfamiliar with parametric logic and algorithmic modeling.

Moving Beyond: Simplification

Another area for improving upon current methods is design space simplification. Although the catalog interface has clear advantages, a main disadvantage is its propensity to overwhelm users with choice fatigue. Especially when numerous variables and multiple design objectives are considered, it can be burdensome to sift through all available data for a given design. In many cases, this procedure may include hundreds or thousands of potential options, which might only be subtly different visually (Aish and Woodbury 2005). Taking a cue from Big Data, these challenges can be addressed by creating tools to extract themes or patterns within a design space. Methods for finding these patterns come from statistics, machine learning, and other fields focused on drawing meaningful conclusions from large volumes of data. Recent examples of architectural applications include dimensionality reduction for visualizing design spaces (Harding 2016a) and the creation of associative archetypes for designing layouts (Derix and Jagannath 2014). Such data analysis techniques can be used to organize and pare down existing catalogs, or set up more meaningful design spaces in the first place.

One specific method is to create categories, groups, or otherwise partition the design space using data clustering. For example, Sicilia et al. (2011) created a categorized database of affordable housing designs based on self-organizing maps and k-means clustering. Such thinking could also be applied to live, performance-based exploration. Figure 6 shows two ways in which clustering can help a designer quickly gain an understanding of the performance of possible designs. In this image, the history of an optimization run was used to cluster a simply supported eleven bar truss. The clustering is based on the design vector of

![A clustered design space for an eleven bar truss. The first row shows the history of an optimization run, which exhibits some patterns but is still visually complex. The second row breaks the data into clusters, which are then sorted by performance. These groups can be used to set new bounds for cluster-based exploration (row 3).](image-url)
each design, rather than performance, meaning that solutions that belong to the same cluster should visually resemble one another. The clusters are then combined with a user-defined flexibility rating to create a more focused design space boundary for each cluster, in which a designer can more quickly explore a region that likely contains higher performing designs. Initially, the designer has only one large design space with a limited sense of regions or performance, as visualized by the fully black design space plot. After running the clustering, a designer can cycle through clusters, focus on the most desirable design family, and either pick an existing design or pursue further exploration in a region of the design space defined by that cluster.

The design space can be similarly organized by performance. The concept of isoperformance, which refers to a set of equivalently performing designs, was developed in aerospace engineering (De Weck and Jones 2006) but can be applied to architectural design as well. Although this categorization is blind to the visual resemblance between similar designs, it greatly enhances design decision-making by presenting a design set in which the designer has almost total freedom, since the performance implications are relatively flat for the selection. A similar design condition results from multiple quantitative objectives that trade off with one another, since an optimization procedure can only return a set of equivalently optimal designs, and a user must define their priorities while selecting the best one. Isoperformance, post-Pareto analysis, and similar a posteriori techniques, which may be applied to previously generated designs in a catalog, ensure that a designer zeroing in on a few designs in a catalog does not miss related areas of the design space.

To illustrate the value of performance-based organization of design spaces, the authors return to the trussed canopy problem. For this problem, the amount of variables has been reduced based on an effect threshold to only include ones that significantly dictate performance. In Figure 7, the designs have been organized in rows that perform similarly to one another. The provided scores represent the amount of structural material required, with 1.00 referring to the initial design, and the parallel coordinate plots visualizing the design vectors for each solution. Although the variables of the problem were selected by the computer based on performance, the diversity of potential designs shows that creative designers still have significant freedom to explore the geometry. Organization by isoperformance quickly informs the designer of which types of designs perform the best. Although many of the organizational methods mentioned in this section are already being applied today, researchers are still attempting to understand which specific techniques are most powerful for early stage design. Future simplification techniques based on data analysis will ideally harness computational speed and power to reduce choice fatigue and quickly direct designers towards high-performing designs without ignoring important details.

Moving Beyond: Visualization

Digital environments have revolutionized the display of information across disciplines. Initial predictions by the pioneers of architectural visualization have largely come true. Paper hand
drawings first gave way to computerized digital documentation, and the field is currently moving towards immersive, VR environments. One branch of architectural visualization focuses primarily on how a building will look, but a quantitative description of how it will behave is increasingly desirable and possible. Since data visualization is a highly developed field, innovations that move beyond the current capability of early-design software will likely fall in two categories: new ways of interacting with data, and new types of graphs or modes of organizing data visually. One promising area for the former is web-based tools (Joyce 2015), such as Design Explorer (Thornton Tomasetti), which visualizes images of user-generated designs along with plots of performance, or more specific dashboard tools (Mazumdar, Petrelli, and Ciravegna 2014) such as UMI Dashboard, which was developed specifically to interact with urban-scale simulation files. A similar data-driven approach is taken by the Urban Interface group and collaborators at KPF (Doraiswamy et al. 2015), and used in structural design by Loos, Verbeeck, and De Laet (2016).

More generally, designers are able to access multipurpose data visualization tools such as Tableau. Other possible areas for improving how data is organized visually include new approaches to illustrating high-dimensional design spaces, condensing large volumes of data into the most relevant details for performance, and mapping the progression of designs and decision-making process for review. Together, the combination of interactivity and novel data organization can augment and shift the criteria for many early design decisions.

REFLECTION AND CONCLUSIONS

This paper has identified four categories of innovations that can improve upon the current procedure for catalog-based design: interaction, automation, simplification, and visualization. While some innovations within these classes retain the catalog interface, others emphasize entirely new modes of engagement with computers during the design process. As simulations of building performance become better integrated into early-stage design, research should move the field of architecture towards more effective human–computer collaboration, even during the initial brainstorming phase. Such collaboration can harness the quantitative power of computational processes along with the creative capabilities of architectural designers to create workflows that respond to the complex demands of contemporary design.

REFERENCES


Designing with Data: Beyond the Design Space Catalog Brown, Mueller


**IMAGE CREDITS**

All images by the authors.

**Nathan C. Brown** is a PhD candidate in Building Technology at the Massachusetts Institute of Technology, where he earned an SMBT degree in 2016. His research seeks to understand how structural considerations interact with other architectural performance criteria in conceptual design. Nathan earned a BSE in Civil and Environmental Engineering at Princeton University, where he studied both architecture and structural engineering, and has worked on building energy retrofit projects for Elevate Energy in Chicago. He was the recipient of the 2016 Structural Engineering Travel Fellowship from the SOM Foundation.

**Caitlin T. Mueller** works at the intersection of architecture and structural engineering. She is currently an Assistant Professor at the Massachusetts Institute of Technology’s Department of Architecture and Department of Civil and Environmental Engineering, in the Building Technology Program, where she leads the Digital Structures research group. Professor Mueller earned a PhD in Building Technology from MIT, an SM in Computation for Design and Optimization from MIT, an MS in Structural Engineering from Stanford University, and a BS in Architecture from MIT, and has practiced at several firms, most recently at Simpson Gumpertz & Heger in Boston.