Suggesting Design Directions: Early Examples of Simulation-Based Guidance for Common Model Types

Nathan C. Brown

Pennsylvania State University
University Park, PA, USA
ncb5048@psu.edu

ABSTRACT
Simulation data can inform early design, but for many design spaces, there is a need for distilling large amounts of performance data into guidance that can support creative, interactive design. Automated computational processes could help support this activity. This paper demonstrates a procedure for finding suggested design directions based on automatic simulations of design options for three common model types. The workflow involves reading in a static geometry and basic simulation information, automatically generating dummy variables, running a series of simulations, applying data analysis to find a direction for improvement, and then returning that direction to the user. This process is demonstrated on a 2D truss for reducing weight, a 3D surface structure for increasing stiffness, and an urban neighborhood concept for increasing PV potential. These simulations reveal one possible foundation for a future design system that intelligently suggests areas of performance improvement for an initially fixed truss, shell, or urban form.

Author Keywords
Parametric design; design suggestion; improved performance; canonical correlation analysis; trusses; surface structures; urban massing

ACM Classification Keywords
I.6 SIMULATION AND MODELING; I.2 ARTIFICIAL INTELLIGENCE; J.6.1 Computer-aided design (CAD)

1 INTRODUCTION
Early stage architectural design involves creative processes in which geometry and other design choices are proposed, discussed, and evaluated. Often, this is done collaboratively among people with different preferences, intentions, expertise, and goals. Such a dialogue can produce useful exchanges in which a solution is proposed, and experts with specific concerns and knowledge suggest a way to improve the given design. This guidance might involve modifications to a system, a set of components, how they are to be arranged, or other decisions that must be made. For certain buildings and related structures, discussion can be primarily about geometry, which has the potential to move or morph continuously within a chosen system. For example, an experienced structural engineer might suggest a more efficient shape than the initial concept, or a building scientist might propose a shifted massing or orientation that reduces energy loads or increases access to daylight.

While these conversations often depend on the experience, knowledge, and creativity of those involved, there is potential for computer simulations to play a role in making suggestions. To start, simulations can quantify the impact of recommended design modifications. Beyond this, computers themselves might find areas of improvement that are novel or counterintuitive for conceptually complex geometries. In other areas of life, numerous recommendation systems exist that operate in a related way, suggesting new songs or products a user might like to purchase. Yet these systems operate on extensive datasets involving many people and their actions. Researchers in architecture and urban design are considering ways to build up similar datasets in the design field [15] and finding other methods for computationally responding to preferred design ideas [4]. Both approaches could have significant future benefits.

However, this paper takes a targeted approach to finding an immediate direction for design improvement. The strategy starts with an automatic parameterization and simulation-based exploration of nearby designs, and then finds trends or patterns in a dataset generated based on these designs. Ideally, a fully formed artificial intelligence system for building design could act like a general suggestion engine—the designer likes "this" geometry; would she or he consider trying "that" geometry, which may improve performance? Or at least try moving in that direction? While not a full suggestion system, this paper considers ways in which three typical geometric types might be evaluated to generate a performance-based direction for design improvement. Through three case studies, it provides examples of meaningful suggested geometric modifications that arise from analyzing a dataset originating in the design itself.

2 BACKGROUND
Today, design firms are increasingly using parametric simulation to generate possible outcomes, determine their performance, weigh their merits and drawbacks, and ultimately make decisions [9, 13, 18]. The idea of performance here refers to quantifiable, desirable aspects of buildings, often in the domain of structure, energy, daylighting, and acoustics. Many common simulation

SimAUD 2020 May 25-27, Online

engines are connected directly to generative design software, making the digital feedback loop between geometry and performance more efficient than ever. As this connection is refined, researchers have developed techniques for generating catalogues of designs, visualizing complex design spaces, finding design space trends, and conducting interactive optimization [3, 12, 19, 20].

Despite the increased prevalence of parametric simulation in architectural practice, there remain unresolved issues. One is that designers must often put forth considerable effort to code a parametric logic, shape grammars, or other strategy to generate possibilities. This task is frequently reserved for specialists and can become time-intensive and tedious if done repeatedly for varied design concepts and projects. To relieve burdens related to coding parametric options, researchers have attempted to create parametric models that generate themselves [10], or domain-specific models that can gradually learn how to predict performance from underlying structures without extensive new simulation [23].

Specific to geometry, some have also created generalized workflows for automatically processing geometry and connecting it to performance outcomes, including for building energy [6], and for views and other urban criteria [7]. Disciplines outside architecture have done considerable work in shape morphing, as it is fundamental to computer graphics and increasingly useful in engineering design [8, 16, 17]. Much of this work contains sophisticated geometric manipulations that improve the design process, but there are still opportunities for removing tedium in computational workflows through targeted automation. This paper introduces one such application, which is suggesting design directions based on performance without the need for coding a parametric model specific to the problem being studied.

Second, the presence of a large design space does not necessarily provide guidance [12]—it is up to the user to interpret the possibilities and make decisions. In many cases, optimization can fill this role of guidance, finding a high-performance design within a user-coded parametric design space [21]. Yet, there are plenty of instances in practice in which a design team generates a conceptual geometry but wants a computer to answer a simple question—from a design that is already preferred, what is a direction that could improve the design in terms of performance? How should the geometry be morphed, and by how much would this adjustment improve a given metric? This paper offers a computational strategy for addressing these questions using automated parameterization, simulation, and analysis, which is especially useful for multi-objective design scenarios or when certain objectives are difficult to quantify.

3 GOALS AND METHODOLOGY

The goal in this paper is to demonstrate a framework for suggesting design improvement on three geometric model types that arise frequently in conceptual parametric design—trussed structures, surface structures, and urban form. The emphasis is on how geometry can be automatically parameterized on granular level, and yet pattern recognition can find smooth, meaningful, performance-based suggested design directions for these geometric typologies. This paper considers structural material quantity, strain energy, and PV potential, but this approach could be used with other quantitative design objectives, as well as with other data science techniques that use an initial dataset to map between geometric variables and a global simulation response.

The general methodology for this workflow is indicated in Figure 1. For each model type, a user must first input a geometry along with basic information necessary for performance evaluation to be implemented. The input geometry can be a preferred design already under consideration for architectural reasons, or an atypical shape for which designers want to know how to improve the performance. The additional input is specific to technical domains—in the examples in this paper, require information includes support locations and loading for the structural case studies. Ideally the input information required beyond geometry is minimal, such that most of the process is automated and users can concentrate on rapidly iterating input geometries or other aspects of the design. However, there are tradeoffs between the specificity of a given evaluation type and its generalizability.

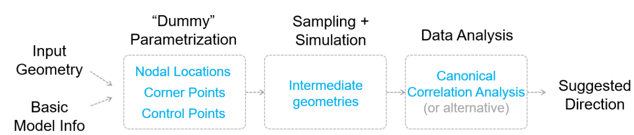

Figure 1. Basic workflow for finding performance-based suggested design directions.

Next, the provided geometry is automatically parameterized with "dummy" variables based on its type: nodal locations for trussed designs, control points for surfaces, and building corner points for urban massing. Since these selections are specific to the model type, they are not generalized further. However, a future "user" could have a library of scripts to choose from and apply the appropriate base parameterization as needed. Some assumptions must be made at this point—for example, how many degrees of freedom should the trusses and control points have, and how far should they be allowed to move? Must building faces stay orthogonal, or should corners be allowed to move independently? In this paper, assumptions are made about each of these questions related to scale, dimensionality, and design freedom. Although in practice such decisions might need frequent updating, certain assumptions could be written once and held constant for, as an example, a firm that frequently works on gridshell roofs of a typical scale.

A series of design samples are then generated, and their performance is simulated. These intermediate design samples represent slightly perturbed versions of the original. As shown in the case studies, the intermediate samples are often not useful designs themselves—they may be wrinkled shells or nonsensical trusses. However, trends can still be found by considering them together as a coherent dataset. To analyze this design space and find a direction for improvement, canonical correlation analysis (CCA) [11] is used. This analysis provides a set of linear coefficients for input variables that maximize correlation with another dataset. Although the primary application of this technique is comparing multidimensional sets of data, when used to find coefficients for dummy design variables that maximize linear correlation with performance, it can produce architecturally compelling directions for improvement. These directions are then returned to the user by way of their coefficients. A 2D visualization of such an outcome is given in Figure 2, showing how two initial parametric variables could be remapped such that a user is moving in a direction that should improve performance.

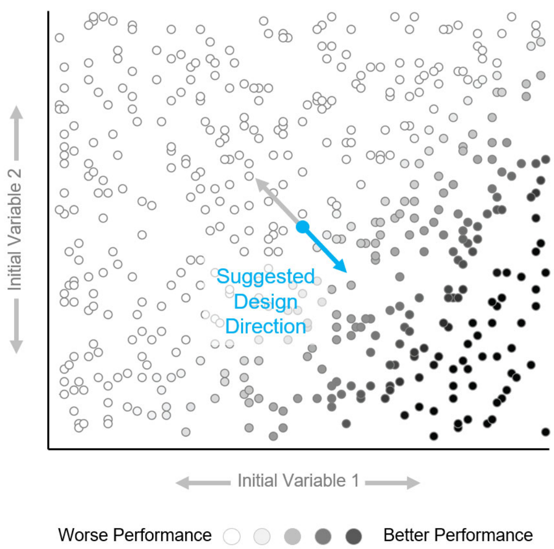

Figure 2. A visualization of how data analysis might find a better design direction than original variables, for a 2D design space.

Although the application of CCA to interactive design has been described before in [2], this paper extends the method to further geometric typologies. Across all model domains, the potential benefits of this approach include finding directions for improvement of a complex geometry that would be difficult to reveal otherwise; finding directions that are counterintuitive; focusing on specific modifications that seem to matter most; and synthesizing desired outcomes through combined analysis of multi-objective guidance.

The current implementation of this workflow uses native Grasshopper components and custom scripts for the automatic parameterization, and the plug-in Design Space Exploration [1] for generating data and manipulating the design. Performance simulations rely on additional plug-ins mentioned in the next section. Once the analysis is completed, designers have access to a slider that morphs the design along the suggested direction. As it is directly on the Grasshopper canvas, this method of suggestion inherits the typical interface and visualization of the parametric design software itself. Although there are currently a few manual steps in the procedure (such as triggering the sampling and data analysis), these could be automated in the future. Although the design manipulation and visualization occur directly in Grasshopper, the functionality for suggesting directions could also be added to a separate interface.

At present, this methodology requires many intermediate simulations to generate suggested directions. Although the sampling and simulation is automated and could be completed while a user is executing different tasks, the time is still significant. Furthermore, this strategy does not provide every way to improve a design, instead giving a single direction per desired quantitative objective (although in a multi-objective scenario, various combinations of priorities could give many directions). Nevertheless, this approach is worthwhile for complicated geometries in which it is difficult to extract a single, cohesive suggested direction without using data. Its relevance would increase for future design environments that run faster simulations and project future directions for multiple design goals simultaneously. The following three case studies demonstrate initial success in finding architecturally meaningful directions of design improvement for basic geometries.

4 CASE STUDIES

4.1 Trusses

In this first example, a script automatically parameterizes a basic truss and then finds suggested directions for improving its performance. Based on the general methodology and interface described in Section 3, this workflow incorporates: 1) reading in a static truss geometry (Figure 3), identifying nodes and creating dummy design variables corresponding to the location of each node; 2) acquiring design information, which includes supports, loads, and boundary conditions; 3)

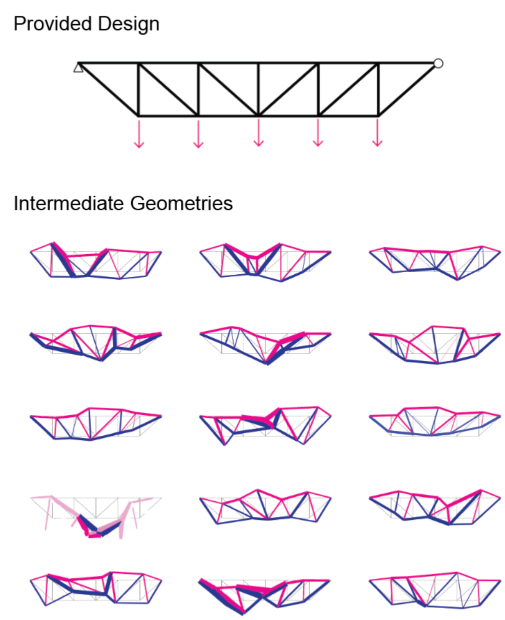

Figure 3. Initial input geometry and a selection of generated intermediate designs for the truss case study. Compressive members are visualized in pink, and tension members in blue.

sampling the initial design space; 4) analyzing the resulting dataset using canonical correlation analysis; and 5) mapping the coefficients back into the original variables to provide a new direction for morphing the structure.

The truss is simply supported and loaded vertically at its lower nodes. For every geometry, Karamba is used to [14] apply loads, calculate internal forces, chose an adequate member size, and return the sized member. Example intermediate designs for the truss are also shown in Figure 3. Very few sample designs are viable solutions, due to the complete freedom of each node to move independently both vertically and horizontally. However, suggested directions for improving the performance of the truss (i.e. lowering its structural material quantity and subsequent weight) seem realistic, as shown in Figure 4. This figure shows the corresponding suggested geometries along with simulation results evaluating the structural weight and deflection in these geometric directions. The three columns represent different amounts of data, which gives a sense of how many simulations are required to generate a coherent result.

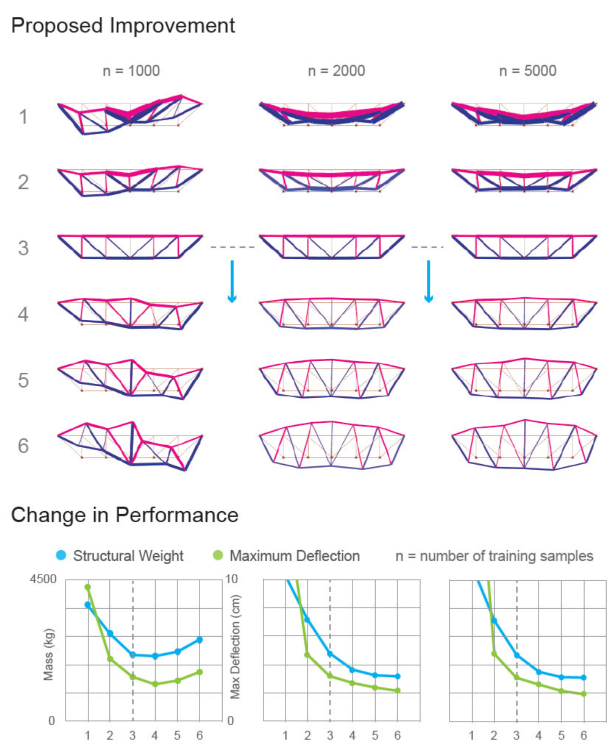

Figure 4. Suggested directions for structural improvement from a flat truss, indicating that increased depth can lower weight.

When asked to find a direction that lowers structural weight, modifications were returned that generally corresponded to truss depth, which tends to control truss efficiency. A reasonably smooth transformation occurred for both the 2,000 and 5,000 sample datasets. This suggests that a fairly large amount of data is needed to extract a discernable pattern, but there are diminishing returns in the smoothness that can be eventually created. Yet these truss directions are encouraging in themselves—no symmetry, constraints, or reasonable bounds were imposed on the design initially, since the raw nodal locations were used directly as trial variables. More information about this approach and case study can be found in [2].

4.2 Surface structures

In the next example, the input geometry is the surface structure shown in Figure 5, and the quantitative design goal is to minimize strain energy. In early design, simulation of this design objective can assess the efficiency of the shape, which has implications for required structural material. The chosen geometry is a loft through a series of curves at different heights and orientations, leading to complex double curvature. Due to BIM and digital fabrication, such geometry is increasingly common for grid shell roofs and other structures around the world. Furthermore, the selection of curves with different heights and curvatures will demonstrate how canonical correlation analysis can highlight specific regions of the surface that require urgent attention for improving performance.

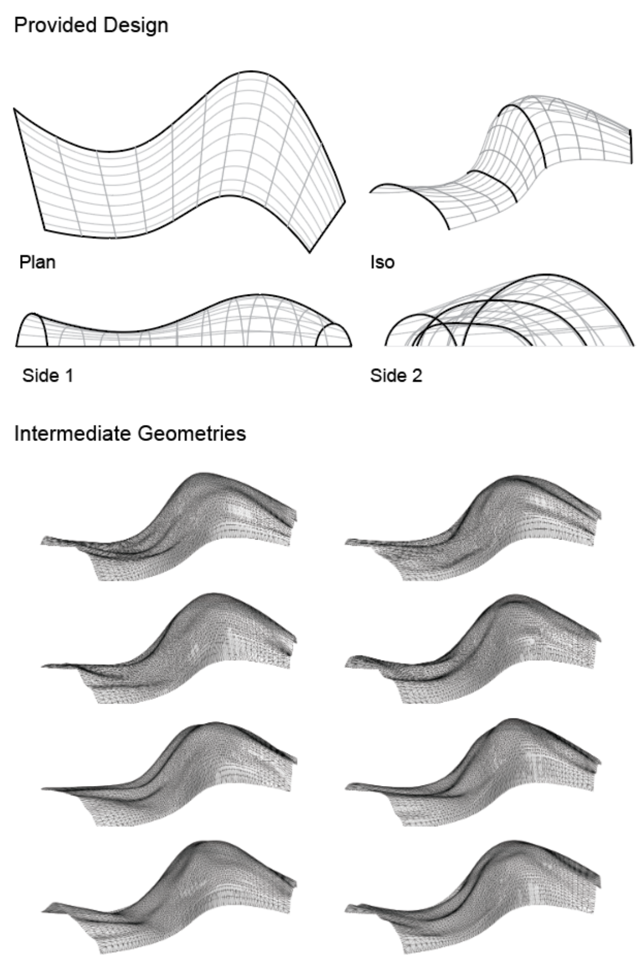

Figure 5. The initial geometry and corresponding samples of intermediate, automatically generated surface structures. The intermediate structures are not smooth, yet they lead to a rational suggested improvement to the original structure.

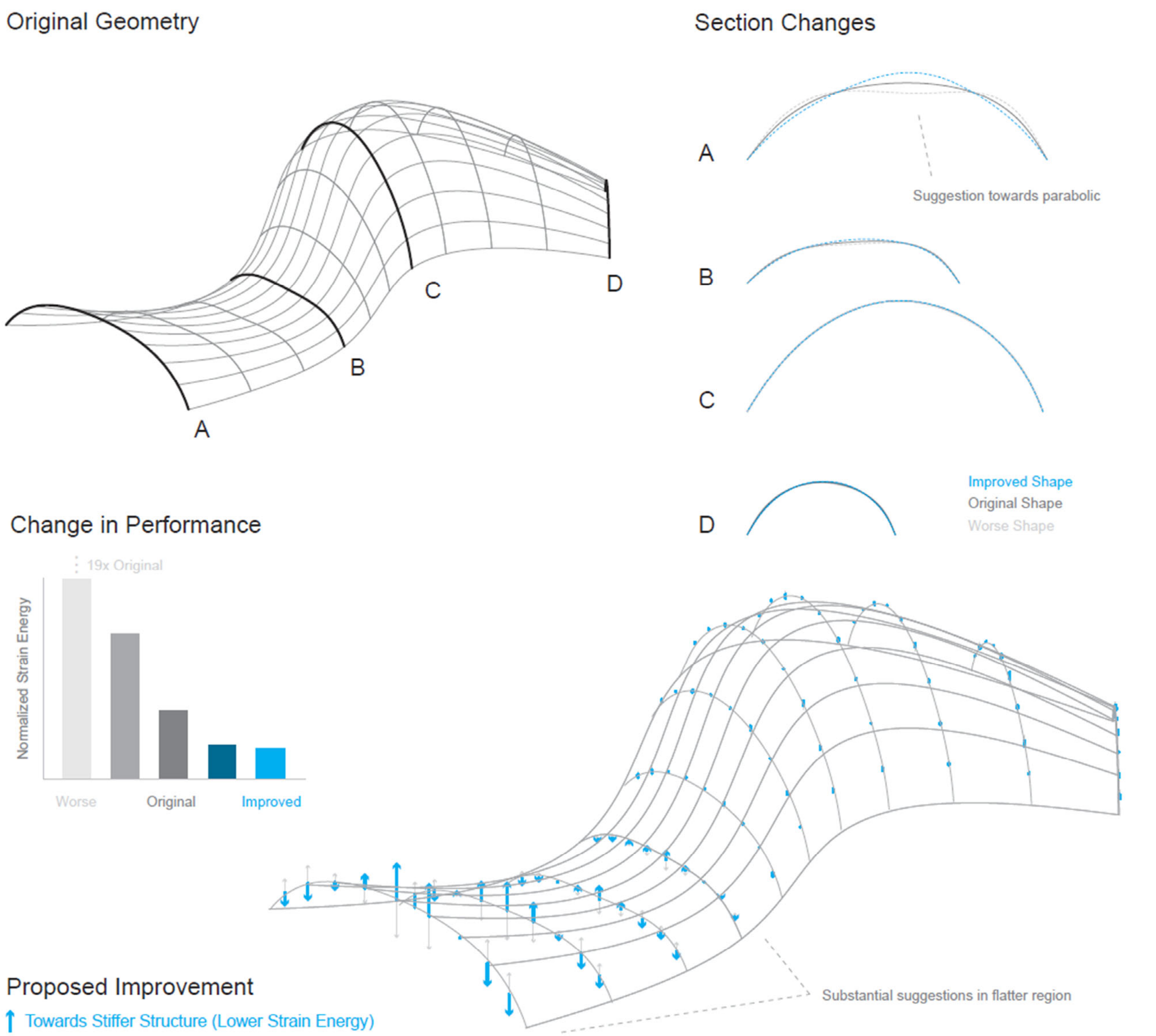

Figure 6. The original surface structure geometry, suggested improvements, and change in performance for these improvements.

The initial shape is first fed into a script that automatically locates control points for manipulation. In this case, 121 control points were located along the surface, and were given a degree of freedom in the z-direction. Next, 10,000 samples were generated based on this new parameterization, and a simulation determined their strain energy. Karamba was again used for the structural simulation, with a uniform load in the negative z-direction. Any edge of the surface touching the bottom plane was assumed to be translationally fixed. The script created for this paper is generalizable to any input surface for which a vertical load and supported edges can be assumed and the software can adequately find control points. However, unlike the truss, there are no set nodal locations for the control points. Thus, for future testing, the resolution of control points might need to be modified to generate meaningful results. The structure is approximately 50 m long, with a maximum span of ~20 m between supports.

Next, a canonical correlation analysis was conducted to determine coefficients that map a direction to improve the performance of the structure. This direction (and its opposite) are visualized in Figure 6, in which the length of blue arrows shows how the structure should be modified to reduce its strain energy, making it stiffer against the vertical load case. By observation, this technique concentrates on the shallower part of the roof, where there is a structurally problematic flat zone. The suggested modification is to morph the curvature of certain sections towards a parabola, which is more efficient for the load case. Although this modification is visually subtle, moving to the structure visualized in Figure 6 cuts the strain energy in half for the entire design, as shown in the normalized graph. It is again notable that although the intermediate samples taken represent wrinkled, unrealistic structure, the overall analysis yields a relatively smooth suggested transition.

It must be acknowledged that when working with surface structures, there are sophisticated formfinding methods to directly arrive at a more efficient structure. In many applications, the force density method, dynamic relaxation, or another technique could find a better shape from the beginning. Nevertheless, there are situations in which an architectural designer might begin with a complex initial form based on their own design goals, which may relate to constructability, competing performance objectives, or another desired quality. In these cases, a suggested direction like this example might show an avenue to improve performance while still staying true to designer intent, or initiate a conversation about tradeoffs between structural performance and preferred geometric outcomes.

4.3 Urban massing

The next case study demonstrates the potential of this automated procedure at the urban scale. In this case, the designer provides the computer with an initial massing for an urban building complex with varying floorplates and heights. The goal is to find a direction for modifying the geometry that would increase the PV potential of the roofs, assuming they would be covered with solar panels. This methodology could be used for urban daylighting or energy simulations, although at present a PV calculation runs considerably faster. A similar procedure involving dummy parameterization, sampling, and analysis is used to determine geometric transformations that correlate with increased PV potential. The initial variables automatically assigned are the corner points of buildings, with the script assuming that buildings can grow taller but remain rooted on the ground. The simulations for PV potential were conducted for the Boston climate using Archsim [5]. The resulting direction is given in Figure 7, which shows changes to the design that affect the overall appearance and spatial sequence of the buildings.

Some aspects of this geometric transformation show that the urban massing workflow is the least developed, as it is a more complex problem requiring additional assumptions in the script if designers require a clean final output. A more refined artificial intelligence process will eventually include constraints on what is feasible in a real urban setting, and build smart rules and capabilities into the design process, such as the ability to pick which walls should stay frozen and recognize that buildings should be combined when they overlap.

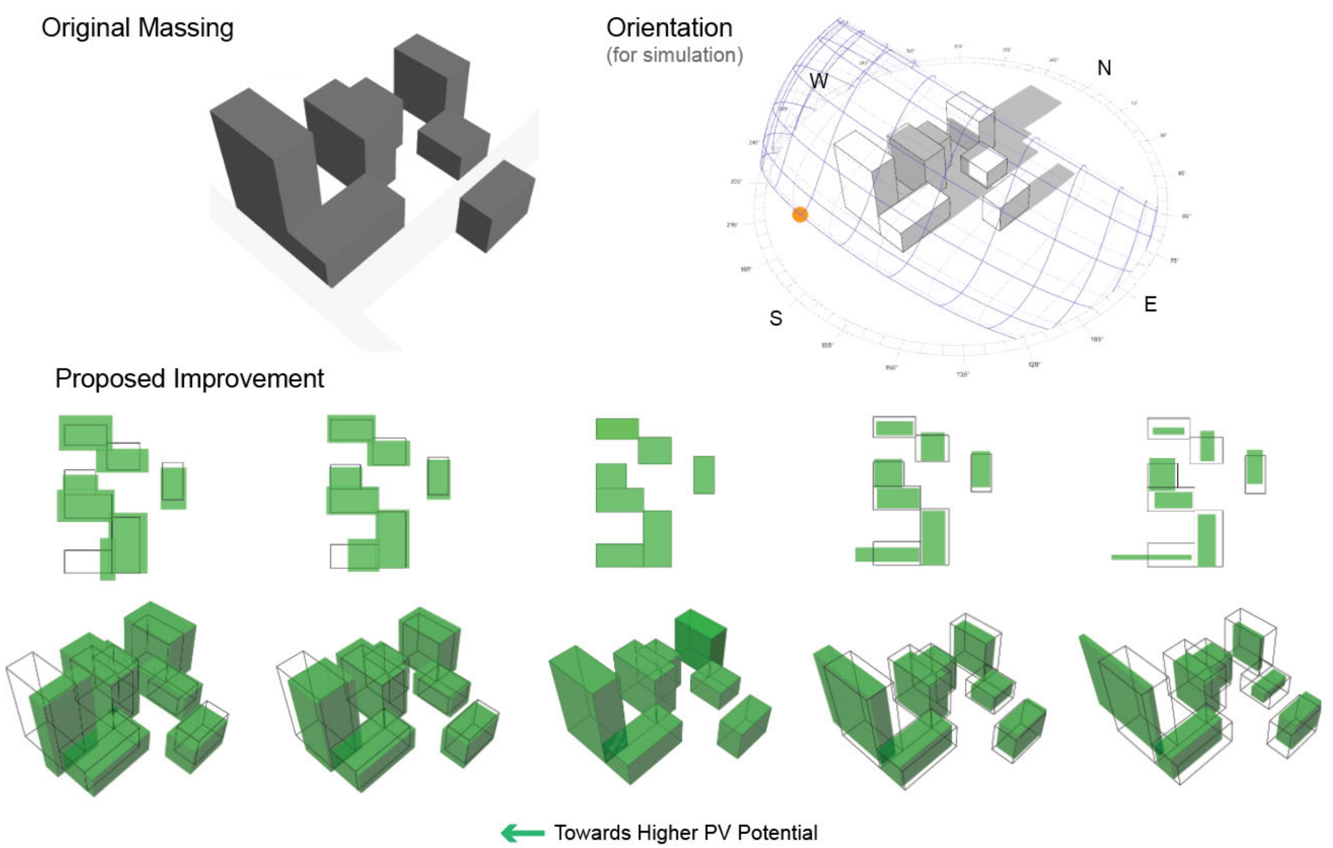

Figure 7. The original massing, simulation orientation, and proposed improvement for increasing PV potential across all building roofs.

Still, the results indicate potentially compelling directions for design exploration. For example, the direction for increasing PV potential seems to show that the tall, southernmost building, which significantly blocks the others in this orientation, should be peeled away, while most other buildings should spread out to maximize surface area. At the urban scale, such suggestions may be more useful than building-by-building adjustment due to the geometric complexity of separate buildings. Consequently, they can provide a starting point for urban designers for where to move next beyond simulation feedback. Such guidance would likely be generated for multiple competing interests in the design, providing a rich exploration of both the design and objective spaces.

5 DISCUSSION

Figure 8 gives a visual a summary of the parallel approaches to providing a design improvement path for the three geometry types. Though these methods show initial success in generating meaningful suggestions for design, a discussion of applications is provided here. It should be mentioned that this automated suggestion workflow should not be the only step in design exploration. As Wortmann [22] notes, simplifying a proeblem into a single (or few) directions for improvement may miss alternatives that are higher performing. A design space might need to be expanded, contracted, or modified throughout the process, and design space exploration or optimization-based workflows are often suitable for this task.

Thus, this workflow is not a replacement for the approaches to data-driven design that have been previously established. However, for applications in which designers want to specifically consider desirable geometries and have a sense for how they might be improved, or control the tradeoff between affinity for the original geometry and potential performance gains, such an automated process may have a role. These situations could include time-pressured processes in which there is not time for systematic optimization for a single solution, when strong qualitative goals are driving the initial geometry generation, or when it is suspected that a computer might discover a compelling direction for improvement that would be difficult to find intuitively due to model complexity or other factors. Although the examples here are preliminary, the ability to extract smooth transformations from datasets full of automatically generated designs that are not individually useful is encouraging. Future experimentation with advanced machine learning methods might yield even more effective results.

Although currently implemented with problem-specific parametric design scripts, this research imagines a future interface with faster simulation and cumulative information from previous designs, in which a variety of metrics could be queried automatically as need. For example, the designers of a complex gridshell shape could provide a desired geometry and ask how to improve stiffness, reduce structural weight, reduce incident solar radiation, or reduce overall energy consumption, while having control over subtle geometric manipulations that achieve each of these desired outcomes. This interaction might occur before or after a more systematic optimization procedure and could augment conversations around preferred designs and worthwhile changes, whenever these conversations occur in the design process. The preliminary case studies in this paper indicate that automatic parameterization combined with data science can find compelling design suggestions for such situations.

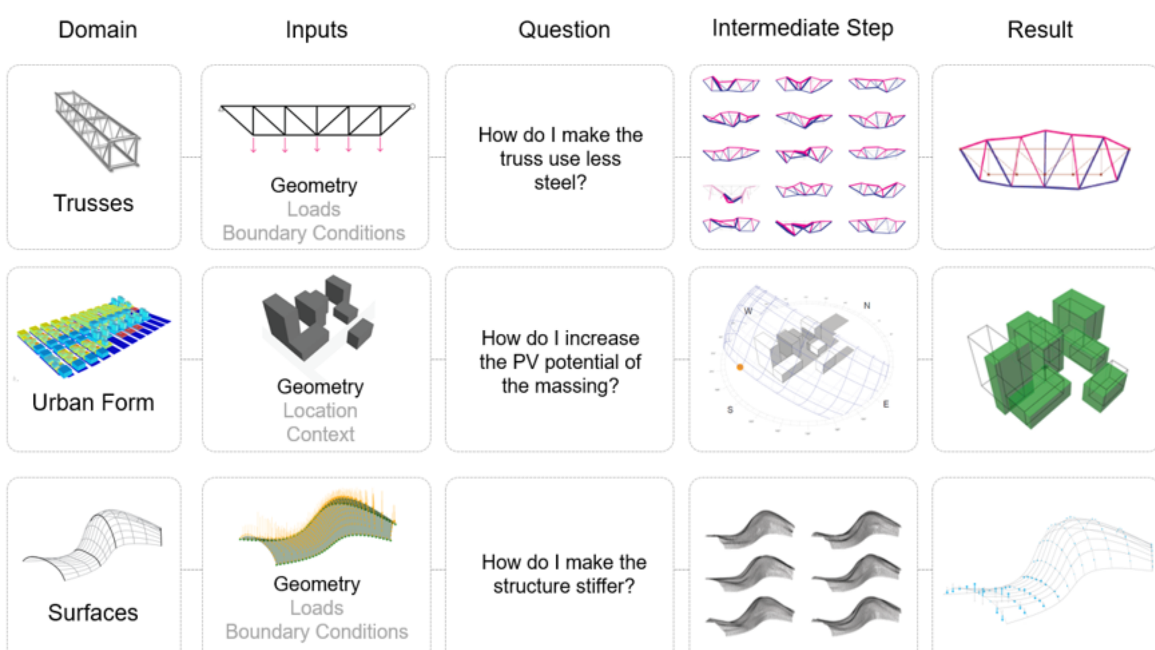

Figure 8. Summary of workflows and suggested directions for the three case studies.

6 CONCLUSION

This paper has demonstrated a computational procedure for generating suggested design directions. This approach includes automatic parameterization, performance simulation, and data analysis, and it was applied to trusses, surface structures, and rectangular buildings. As this workflow is still early, there are many areas for future work. First, rather than using a combination of scripted components and custom code, this could be implemented in its own design interface. A new interface could make additional assumptions for each geometric type, as mentioned for the case studies, to increase its generalizability for input designs. The workflow should also be tested for more complex geometry, additional performance metrics, and other data analysis techniques that go beyond linear correlation. Such a process could also be combined with ongoing research on automatically translating a 2D sketch into 3D models, which would allow designers to literally sketch ideas and get feedback rather than drawing first in CAD. Nevertheless, the initial results in this paper are a step towards intelligent design suggestion based on performance simulation.

ACKNOWLEDGMENTS

The author would like to thank Caitlin Mueller and Digital Structures for their contributions to the software used here.

REFERENCES

1. Brown, N. and Mueller, C. Design Space Exploration, 2017. www.food4rhino.com/app/design-space-exploration.
2. Brown, N. and Mueller, C. Gradient-based guidance for controlling performance in early design exploration. *Proceedings of IASS*, (2018), Cambridge, MA.
3. Brown, N. and Mueller, C. Design variable analysis and generation for performance-based parametric modeling in architecture. *International Journal of Architectural Computing 17*, 1 (2019), 36–52.
4. Chatzikonstantinou, I. and Sariyildiz, I.S. Addressing design preferences via auto-associative connectionist models: Application in sustain architectural Façade design. *Automation in Construction 83*, (2017), 108–120.
5. Dogan, T. 2014. ARCHSIM. www.solemma.com.
6. Dogan, T. and Reinhart, C. Shoeboxer: An algorithm for abstracted rapid multi-zone urban building energy model generation and simulation. *Energy and Buildings 140*, (2017), 140–153.
7. Doraiswamy, H. et al. Topology-based catalogue exploration framework for identifying view-enhanced tower designs. *ACM Transactions on Graphics 34,* 6 (2015), 230.
8. Gao, L. et al. A data-driven approach to realistic shape morphing. *Computer Graphics Forum 32*, 2.4 (2013), 449–457.
9. Gerber, D.J. et al. Design optioneering: multi-disciplinary design optimization through parameterization, domain integration and automation of a genetic algorithm. *Proceedings of SimAUD,* (2012), 1–8.
10. Harding, J. and Shepherd, P. Meta-parametric design. *Design Studies 52*, (2016), 73–95.
11. Hardoon, D.R. et al. Canonical Correlation Analysis: an overview with application to learning methods. *Neural Computation 16*, 12 (2004), 2639–2664.
12. Mueller, C.T. and Ochsendorf, J.A. Combining structural performance and designer preferences in evolutionary design space exploration. *Automation in Construction 52*, (2015), 70–82.
13. Oxman, R. Performance-based design: current practices and research issues. *International Journal of Architectural Computing 6,* 1 (2008), 1–17.
14. Preisinger, C. 2014. Karamba. www.karamba3d.com.
15. Reynolds, D. et al. Structural design taste test. *Proceedings of IASS*, (2018), Cambridge, MA.
16. Schulz, A. et al. Retrieval on parametric shape collections. *ACM Transactions on Graphics 36,* 1 (2017), 1–14.
17. Sederberg, T.W. and Parry, S.R. Free-form deformation of solid geometric models. *SIGGRAPH '86 20*, 4 (1986), 151–160.
18. Shea, K. et al. Towards integrated performance-driven generative design tools. *Automation in Construction 14*, 2 (2005), 253–264.
19. Turrin, M. et al. Design explorations of performance driven geometry in architectural design using parametric modeling and genetic algorithms. *Advanced Engineering Informatics 25*, 4 (2011), 656–675.
20. Wortmann, T. 2017. Surveying design spaces with performance maps: a multivariate visualization method for parametric design and architectural design optimization. *International Journal of Architectural Computing 15*, 1 (2017), 38–53.
21. Wortmann, T. and Nannicini, G. 2017. Introduction to Architectural Design Optimization. In *City Networks – Collab. And Plan. for Health and Sustainability,* (2017), 259-273.
22. Wortmann, T. and Schroepfer, T. 2019. From Optimization to Performance-Informed Design. *Proceedings of SimAUD,* (2019), 261–268.
23. Xuereb Conti, Z. and Kaijima, S. A flexible simulation metamodel for exploring multiple design spaces. *Proceedings of IASS*, (2018), Cambridge, MA.