Increasing applications of parametric design and performance simulations by architectural designers present opportunities to design more resource- and energy-efficient buildings via simulation-based optimization. But Architectural Design Optimization (ADO) is less widespread than one might expect, due to, among other challenges, the problematic integration of optimization with architectural design. This problematic integration stems from a contrast between “wicked” or “co-evolving” architectural design problems and optimization problems. To mitigate the contrast between architectural and optimization problems, this paper presents Performance Explorer, an interactive, visual tool for performance-informed design space exploration (DSE).

Performance-informed DSE emphasizes selection, refinement, and understanding over finding highest-performing design candidates. Performance Explorer allows interactive DSE via a visualization of a fitness landscape, with real-time feedback provided with a surrogate model. Performance Explorer is evaluated through a user test with thirty participants and emerges as more supportive and enjoyable to use than manual search and/or optimization.

**Keywords:** Architectural Design Optimization, Performance-informed Design, Interactive Visualization, Design Tool

**INTRODUCTION**

The increasing use of computational methods such as parametric modelling, performance simulations, and optimization raises questions about how to integrate such methods into architectural design processes, which are characterized by wicked problems and the co-evolution of design problems and solutions (Johnson 2017).

This paper presents Performance Explorer, a visual and interactive tool for performance-informed design space exploration (DSE) (figure 1). DSE is an active field of research devoted to supporting designers in selecting from the large numbers of design candidates represented by parametric models and in understanding the underlying design spaces (e.g., Shireen et al. 2017). **Performance-informed** DSE considers design spaces relative to one or more quantitative performance objectives, such as environmental...
Figure 1
Performance Explorer in Grasshopper. From left to right: (a) Morphology (i.e., appearance) of the current design candidate, (b) definition of the parametric model in Grasshopper (note the green box with number sliders representing design parameters and their current values), and (c) the Performance Explorer window.

or structural performance.

Design spaces with performance dimensions are also known as fitness landscapes. Surrogate models interpolate fitness landscapes from a sample of design candidates with known performance, using a variety of statistical and machine-learning methods (Koziel and Leifsson 2013) (figure 2).

Performance-informed DSE contrasts with concepts such as performance-driven design (Shea et al. 2005), performative or performance-based design (Oxman 2008), and generative design (Nagy et al. 2017) that propose to automate the search for high-performing design candidates, most prominently through optimization methods such as the popular genetic algorithms (Wortmann and Nannicini 2017). Performance-informed DSE emphasizes supporting designers with selecting and refining promising design candidates and with understanding fitness landscapes, and aims to better integrate optimization into architectural design processes.

This paper provides a brief background to performance-informed DSE, introduces Performance Explorer, and presents results from a user test. The user test compared the interactive DSE afforded by Performance Explorer to manual DSE (i.e., manipulating parameter values by hand) and automated DSE (i.e., using an optimization tool).

BACKGROUND
Selection entails presenting designers with a meaningful choice from a set of well-performing solutions rather than with a single optimization result. Refinement entails supporting manual parameter changes, either to bring a well-performing candidate closer to a designer’s intent, or to enhance the performance of an otherwise promising candidate. Understanding refers to surveying the entire fitness landscape and analyzing relationships between design parameters and performance.

Approaches that support selection often employ a clustering method to group large numbers of parametric design candidates-resulting, for example, from optimization-into clusters of similar candidates (e.g., Stasiuk et al. 2014). A single design candidate from each cluster then represents the corresponding “type” of designs, which makes it easier to select a design candidate for further development.

Compared to selection, supporting refinement is relatively straightforward. Allowing the intuitive adjustment of individual design parameters is a basic feature of visual programming platforms such as Grasshopper. Surrogate models-approximations of fitness landscapes that are much faster to compute than simulations-provide designers with real-time performance feedback to such adjustments (e.g.,
Geyer and Schlüter 2014). The interactive performance maps proposed by (Wortmann 2017) not only use surrogate models for real-time feedback, but also to visualize (a projection of) the entire fitness landscape (figure 3a). In this way, designers can see what kinds of parameter adjustments might lead to better performance. Performance maps also support selection by representing individual design candidates relative to the positions of other design candidates in the fitness landscape.

Exploring the effects of parameter changes on performance in real-time helps designers to also grow and refine their understanding of fitness landscapes. The real-time feedback by (Geyer and Schlüter 2014) and the interactive performance maps by (Wortmann 2017) not only support adjustments in local regions, but also global explorations that lead to a better understanding of fitness landscapes. Performance maps support this exploratory analysis with a visual overview of fitness landscapes.

PERFORMANCE EXPLORER INTERFACE
This paper presents Performance Explorer, an interactive visualization of fitness landscapes based on surrogate models. Performance Explorer has three critical functions: (1) It provides an (approximated) overview of the entire fitness landscape, (2) allows the interactive and targeted enhancement of the underlying surrogate model, and (3) provides multiple representations of a parametric design candidate—as a 3D model of its appearance, i.e., morphology, as a dot on the Performance Map, and as a radial plot of its parameter values—to promote understanding and creativity (Yamamoto and Nakakoji 2005).

Performance Explorer is a plug-in for the parametric design software Grasshopper (figure 1a). To use Performance Explorer, one first runs Opossum—a model-based optimization tool in Grasshopper—for several function evaluations (i.e., simulations). During this optimization phase, Opossum samples the design space and constructs increasingly accurate surrogate models (Wortmann 2018).

The Performance Explorer window has four main GUI (graphical user interface) elements: The performance map, the performance scale, the variable plot, and the Simulate and Refresh buttons (figure 3).

Performance Map
The performance map (figure 3a) is a two-dimensional, radial mapping (i.e., projection) of the high-dimensional fitness landscape, insofar as Opossum has explored it during the optimization phase. Every dot represents a design candidate whose performance value Opossum has simulated. The remaining colored area represents design candidates whose parameter values the performance map has interpolated via barycentric coordinates and whose performance values Opossum has approximated with a surrogate model (Wortmann 2017).

The six radial axes indicate the radial mapping and are labelled according to the names of the number sliders, i.e., the parameters, in Grasshopper (fig-
Designers interact with the performance map via the white *position cross*. The position cross indicates the position of the current Grasshopper model (i.e., of the parameter values) in the fitness landscape (figure 3a). When Performance Explorer starts, or the performance map is regenerated, Performance Explorer (re-)sets the position cross to the best-known design candidate.

Designers can change the current parameter values by dragging the position cross across the fitness landscape, which resets the slider values in Grasshopper, and thus the parametric model. Since Performance Explorer derives these parameter values by interpolating between simulated design candidates, they change non-linearly, even when designers drag the position cross along one of the coordinate axes.

Alternatively, designers move the position cross by manipulating parameter values not on the performance map, but via the variable plot (figure 3c) or with Grasshopper’s number sliders. This *direct* manipulation allows more exact changes of individual parameter values. Changing a single parameter moves the position cross along a single axis, which helps to better understand the visualization.

**Performance Scale**

The performance scale (figure 3b) provides a legend for the colors of the performance map. The white...
“performance bar” and the number next to it represent the (simulated or approximated) performance value corresponding to the current position of the position cross.

In this way, Performance Explorer accompanies designers’ parameter changes with real-time performance values. To achieve this real-time feedback, Performance Explorer temporarily disables the performance simulation in Grasshopper-Grasshopper otherwise responds to parameter changes with triggering new, potentially time-intensive simulations—and displays performance values that are known or approximated from the surrogate model.

**Variable Plot**
The variable plot (figure 3c) is a radial plot of the current design candidate’s parameter values, colored with the corresponding performance value. Since it is hard to visually estimate parameter values from the performance map, the variable plot supports designers’ intuitions with an alternative visualization of the current design parameters and (approximated) performance value. The axes of the variable plot correspond to the “main” axes of the performance map, but with radial coordinates instead of a radial mapping (i.e., the intersections between the variable plot and the axes represent individual parameter values). Designers can refine individual parameter values by dragging corners of the variable plot.

**Simulate and Refresh Buttons**
The *Simulate* and *Refresh* buttons (figure 3d) implement two critical, novel features of Performance Explorer, whose combination allows designers to manually enhance surrogate models that underlie the performance map.

Performance Explorer disables the performance simulations of parametric models in Grasshopper to allow real-time performance feedback, especially for time-intensive simulations. Avoidance of simulations means that, for design candidates that have not been simulated previously, the performance values indicated by the performance map are only estimates.

Designers use the Simulate button to verify the performance values for promising design candidates. When a designer presses the simulate button, the performance bar jumps to the “correct” (i.e., simulated) value, with larger jumps indicating a less accurate approximation from the surrogate model. By repeatedly using this feature, designers can get a sense of the underlying surrogate model’s accuracy. Based on the author’s experience, these estimates are more precise for design candidates that are predicted to perform better, and less precise for design candidates that are predicted to perform worse.

Importantly, triggering a simulation also implies generating an additional sample for the surrogate model. But performing a simulation does not immediately regenerate the performance map, since this regeneration can take several seconds. Rather, the recalculation of the surrogate model and regeneration of the performance map is triggered only when designers press the Refresh button, after performing one or more simulations. Such a regeneration can result in visible changes to the performance map, which typically indicate areas of the fitness landscape where the surrogate model’s accuracy has improved. Regeneration also resets the position cross to the best-know (simulated) design candidate, which potentially has changed. The remainder of this paper presents the methodology and results from a user test of the efficacy of the ideas behind and implementation of Performance Explorer.

**USER TEST METHODOLOGY**
The user test investigated the hypothesis that, by allowing designers to introduce qualitative criteria into their search for quantitatively well-performing design candidates, Performance Explorer supports performance-informed DSE more than either manual search or automated search (i.e., optimization).

Thirty subjects—students or researchers in architecture with at least some familiarity with Grasshopper—participated in the user test. Participants used their personal laptops for the test.
**Design Task**

The user test involved the following, performance-informed DSE task:

*For a given parametric model, find a well-performing design that is a promising starting point for further architectural development.*

This definition of the design task follows the insight by (Bradner et al. 2014) that, in ADO, optimization results are more often used as starting points for further design development than as end results. The parametric model-defined in Grasshopper-represented a small pavilion and had six parameters (three parameters for the height of the pavilion's corners, one for its center height, one for the size of its openings, and one for the depth of the overhangs over the entrances) (figure 4).

The quantitative performance criterion was the pavilion's maximum displacement under dead load. Participants were told to interpret displacement as a relative measure of structural performance and to ignore factors such as the pavilion's thickness or material. Participants were reminded that their task was not to find the pavilion with the lowest maximum displacement, but to find a well-performing pavilion that they considered a promising conceptual design from an architectural point of view (based on individual design criteria, e.g., conceptual, formal, or programmatic ones). This “preferred design candidate” could be best-performing, but not necessarily.

On an Intel i7 6700K CPU with 4.0 Ghz and eight threads, performing the structural simulation takes about 300 milliseconds. This time is short enough for sufficient DSE within ten minutes, but long enough to make the real-time feedback afforded by Performance Explorer meaningful, especially on the participants’ (slower) laptops.

**Performance-informed DSE Methods**

The participants performed the performance-informed DSE task with three distinct methods, for ten minutes each:

1. The *Manual* method involves manipulating the parameter values of the parametric model directly and simulating the resulting design candidates.

2. The *Automated* method involves optimizing the parametric model with an optimization tool (Opossum)-with the number of function evaluations and runs decided by the participants-and choosing a well-performing candidate from a table of results. This method allows “fine-tuning” of design candidates by adjusting parameter values directly.

3. The *Interactive* method involves first running optimization to generate a surrogate model, and then-using Performance Explorer-searching a visualization of the surrogate model (i.e., of the approximated fitness landscape) for well-performing design candidates.

Compared to “pure” optimization that wants to find only a single, best-performing solution, the automated method introduces an element of choice by offering a table of results. This element of choice facilitates a more meaningful comparison of the three methods in terms of performance-informed DSE, since optimization tools that return only one best-performing design candidate facilitate understanding and exploratory divergent thinking only to a small degree. All methods were demonstrated to the participants before the user test.
Experimental Design
Each participant performed the performance-informed DSE task with each of the three methods (manual, automated, and interactive). Participants could use each method for at most ten minutes. This experimental design introduced a potential bias, because participants progressively learned more about the design task’s design space with each method. To mitigate this bias, the order in which they used the methods was randomized.

Data Collection
After using each method, participants indicated the parameter and objectives values of their preferred design candidate, their criteria for choosing it, and their strategy for finding or selecting it. Participants rated in how far the preferred design candidate was a promising starting point for further development and in how far they got a good overview over potential design candidates (i.e., the fitness landscape). Participants also had the opportunity to note any other comments and/or features requests regarding each method. After using all three methods, participants ranked them in terms of how much they supported them with the performance-informed DSE task and in terms of how much they enjoyed using them. They also had the opportunity to record any final comments.

RESULTS
Manual Method
With the manual method, participants gave a wide range of motivations for choosing their preferred design candidates, such as symmetry, shading, “expressing the idea of flight,” “the potential for interesting programs to occur,” “inviting and noticeable entrances,” and ease of construction.

One can identify three DSE strategies employed by the participants: (1) Random manual exploration, i.e., manipulating the parameter values unsystematically, (2) strategic manual exploration, i.e., manipulating the parameter values systematically by setting them in a certain order or by trying to achieve a certain shape, and (3) analytic manual exploration, i.e., trying to understand the impact of the parameters on the design and its displacement.

Responses indicating random exploration included phrases such as “play around [with] the parameters” or “moving the sliders until I find a nice design.” Responses indicating strategic exploration included phrases such as “decide the three heights first and then the center height” or “I immediately looked at stepping the heights.” Responses indicating analytic exploration included phrases such as “I tried to figure out the relationship between center height and deformation” or “I ... tested the impact [parameter changes] made on the overall deflection.”

Two participants noted that, with the manual method, they had “more freedom” than with the others, but another noted that “I didn’t know [how] to improve the ... displacement, without changing my shape.” (Performance Explorer assist in such situations by presenting an overview of the fitness landscape.) Five participants noted that the manual method was more time-consuming than the other methods. One explained that “many interesting shapes can be achieved ..., but I would not have noticed or found them (unless I spend 1,000 hours).”

Automated Method
With the automated method, participants again gave a wide range of motivations for choosing their preferred design candidates, albeit with a stronger focus on structural performance. One participant mentioned that (s)he felt “compelled to compromise my massing shape for its performance.”

Most participants sought a compromise between structural performance and other criteria (e.g., aesthetics, experience, shading, etc.). A typical response from this group was “Aesthetically [the design] looks quite balanced and it has a good optimization value.”

22 participants ran the optimization and then selected a design from the table of results. A typical response from this group was “I ran the [optimization] for a while, then I checked all the possible solu-
tions, and I picked the one following my design criteria among the ones with the best displacement values.” Only three participants from this group “finetuned” their selected design candidates by manually adjusting parameter values. Two participants used the best optimization result as a starting point for further manual exploration, and three simply accepted the best optimization result. The remaining three participants followed idiosyncratic strategies (such as maximizing deflection or trying to stop the optimization when “the design looks interesting”).

Several participants indicated that the well-performing designs in the results list were “very similar” or “do not change too much.” Accordingly, seven participants requested improvements to the results list, such as filtering results according to similarity (e.g., with a clustering method). In summary, the automated method increases the need to balance trade-offs between the quantitative performance objective and additional, qualitative, criteria.

**Interactive Method**

With the interactive method, motivations for choosing the preferred design were similar to the automated method. Twenty participants sought a compromise between structural performance and other criteria, seven participants selected a design for non-structural, largely aesthetic reasons, and three considered structural performance only. But there is evidence that, compared to the automated method, participants approached the trade-offs between different criteria with more deliberation and freedom:

One participant mentioned that the “visualization offers an opportunity to change parameters according to ... reason, not ... experiment;” and another followed his original design concept, but with parameters that “were better controlled to better optimize the maximal deflection.”

Several participants indicated that the interactive method was more permissive in terms of accepting suboptimal performance values. One mentioned that the preferred design was “visually appealing, but not very efficient;” and another that “even if the objective is not very low, the shape of the model is good-looking.” A larger group of participants was very satisfied with the balance struck by the preferred design, for example finding a shape “that has a nice structure and it also suits the design.” The following response expressed the perhaps highest satisfaction:

*This design has the most desired shape with the lowest displacement value. When utilizing [Performance Explorer], I was able to somewhat determine the parameters that I want ... I was able to visualize which parameters will provide me with the most stable design.*

The interactive method received several very positive comments, for example that it was enjoyable to use, “provides more options of the design shape,” “makes a lot of sense;” or that its visualization was “helpful” or even “very very helpful” and allowed for the “immediate pinpointing of ideal tests.”

In summary, the interactive method accommodates a wide range of performance-informed DSE strategies. It allows designers to pursue strategies associated with the manual and automated methods and enhances these strategies with real-time feedback and the understanding afforded by the performance map (i.e., the visualization of an approximated fitness landscape). The following quantitative comparison supports this conclusion.

**Quantitative Results**

After using each method, participants rated in how far their preferred design candidates were promising starting points for further development and in how far the methods provided an overview over the design space. Although some participants were very satisfied with the design candidates they discovered manually, the interactive method was the most effective in terms of supporting the discovery of promising design candidates and providing an overview of the design space. After completing the performance-informed design task with the three methods, participants ranked the methods in terms of their helpfulness with the design task and in terms of their enjoyment in using them.

Differences between the three methods are
more pronounced in the \textit{a-posteriori} comparison (left in figure 7) than in the individual evaluations (right in figure 7). This larger difference is probably due to the participants’ development of a better understanding of potential design candidates (i.e., the design space) and the strengths and limitations of the three methods, as well as the requirement to rank the methods (i.e., participants had to pick a best and a worst method). The interactive method emerges as (1) the most supportive and (2) enjoyable method that results in (3) the most promising starting points for further development and (4) affords the most comprehensive overviews over design spaces.

LIMITATIONS
The user test had three major limitations: (1) The selection of participants, (2) the method of data collection, and (3) the simplified DSE task.

All participants were from academia, and thirteen were undergraduates. This selection allowed a larger sample size but might limit the relevance of the user test for professional practice. On the other hand, many participants had at least some experience with professional practice.

Another limitation was the method of data collection: Participants’ written responses sometimes were unclear or left out important aspects. Although this limitation was partially compensated by the compared to similar studies (e.g., Shireen et al. 2017)-larger number of participants, the interpretation of the participants’ written responses is somewhat subjective. Future studies might gather richer data, for example by recording participants’ explorations, and/or by verbally interviewing them.

The final limitation concerns the performance-informed DSE task itself. Compared to the simulations and parametric models used in architectural practice (Hudson 2010), the design task was simplified dramatically. This simplification was necessary to reduce the user tests’ requirements for software, expertise, and-most critically-simulation time. Nevertheless, the participants’ responses demonstrate that the simplified design task retained sufficient complexity to allow meaningful explorations.

DISCUSSION
Some participants felt “freer” with the manual method, but most participants nevertheless preferred the automated and interactive methods. Some participants mentioned that the latter methods allowed the examination of more design candidates. For more realistic problems with longer simulation times, this advantage would become more pronounced, especially with the interactive method.

Only a small number of participants accepted the “best” design candidates from optimization. Although the experimental design likely encouraged this outcome, it nevertheless confirms design-theoretical ideas that posit optimization as a medium for reflection (e.g., Bradner et al. 2014; Johnson 2017). The comments on the automated method highlight the need to present optimization results in terms of
meaningful differences between design candidates.

Designers applied various (implicit and explicit, quantitative and qualitative) criteria and prefer selection from a range of alternatives over a single, “optimal” solution. Beyond selection, some designers prefer to understand the relationships between design parameters, the morphologies of design candidates, and their performance. In other words, some designers prefer to gain insight into the “black boxes” defined by parametric models and performance simulations. Although Performance Explorer’s interactive visualization of fitness landscapes and multiple representations do not guarantee understanding, they certainly support its acquisition.

CONCLUSION
The user test presented in this paper clarifies that the better integration of optimization into architectural design process requires performance-informed DSE tools with enhanced visualization and interactivity. Performance Explorer undoubtedly can be improved further, but nevertheless represents a valuable proof-of-concept with novel and innovative features. The user test not only validated this approach, but also yielded valuable insights into performance-informed design processes, which can serve as a framework for future research in this direction.

Accordingly, future work includes improving Performance Explorer based on the participants’ feedback and extending it for multiple performance objectives, as well as conducting a more extensive user test. Such a test would employ interviews and screengrabs to further deepen the theoretical framework of performance-informed DSE in terms of selection, refinement, and understanding.

ACKNOWLEDGEMENT
This paper is based on a chapter from the first author’s PhD thesis (Wortmann 2018). Opossum and Performance Explorer were developed at SUTD’s Advanced Architecture Lab and supported by the SUTD-MIT International Design Centre under grant numbers IDG215001100 and IDG2170010.

REFERENCES
Hudson, R 2010, Strategies for parametric design in architecture: an application of practice led research, Ph.D. Thesis, University of Bath
Koziel, S and Leifsson, L (eds) 2013, Surrogate-Based Modeling and Optimization, Springer New York, New York, NY
Oxman, R 2008, ‘Performance-based design: current practices and research issues,’ IJAC, 6(1), pp. 1-17
Stasiuk, D, Thomsen, MR and Thompson, EM 2014 ‘Learning to be a Vault’, Fusion - Proceedings of the 32nd eCAADe Conference, Newcastle upon Tyne, UK, pp. 381-390