Jamie Queisser and Saqib Aziz

Searchfield: Navigating n-Dimensional Design Spaces

Abstract: Algorithmic design systems enable access to a variety of performance simulation tools, allowing to evaluate and explore design variations. Design tasks are often defined through multiple input parameters, thus spanning large high-dimensional solution spaces. The designers are faced with challenges when navigating and comprehending this solution space while striving to meet the requirements given by the design brief and aiming to achieve certification guidelines (BREEAM or DGNB). The described method projects the high-dimensional solution space to two dimensions while preserving similarities between individual solutions, making it intuitive for designers to navigate. The solution space is overlaid with performance estimates to assist the designer during the decision-making process and features methods for selecting and filtering according to design and performance criteria. A design project currently in development by the corresponding authoring firm, has been used to serve as a case study probing the design principles.

Keywords: design space exploration, multivariate visualization, dimensionality reduction, machine learning, sustainable design

1 Introduction

The AEC industry has experienced a rapid digital transformation, replacing outdated design and fabrication processes with advanced digital automation and collaborative platforms (Carpo 2017). Building Information Modelling (BIM) has successfully facilitated real-time information exchange among designers, engineers, and contractors within a unified design framework. Research suggests that decision-making in planning decreases exponentially over time towards later design development stages, emphasizing the importance of early-stage performative design optimization for improved economic and ecological performance of buildings (Kohler and Moffatt 2003). While BIM software has limitations in early design stages, design firms often rely on dynamic Computer-Aided-Design (CAD) modeling software like Rhinoceros3D and Grasshopper3D for form exploration and creation (McNeel 2010).

This research examines how technology can facilitate collaborative exploration of design spaces by designers and engineers, considering interdependencies among different domains. The study also investigates how to communicate decision-making and performative optimization processes in the early design stage in a way that architects can easily relate to. To address these issues, an experimental multi-domain and

https://doi.org/10.1515/9783111162683-029

collaborative design interface is introduced that allows users to collectively navigate through decision-making and solution spaces efficiently.

Unsupervised machine learning (ML) identifies patterns and structures in data without predefined labels or categories (Wang and Biljecki 2022). It differs from supervised learning, which relies on labeled examples for learning, and is used when there is no pre-established target or output variable to be predicted. This paper uses Principal Component Analysis (PCA, unsupervised ML) to project multi-dimensional vectors to comprehensible and user-friendly 2D interactive graphs and Linear Regression (supervised ML) for performance estimation, as described in Sec. 3.3.2 and 3.4.3.

2 Background

2.1 Design Process in Architectural Practice

Characterizing the diverse design processes in architectural practice is a challenging and generalized endeavor. Architectural design tasks are often considered wicked problems, as defined by Rittel and Webber (1973). Each design task, whether it involves the entire building or sub-tasks, presents numerous decisions and perspectives that reflect the unique design culture of an architectural office. The decision-making process is highly iterative, driven by changing criteria and requirements encompassing aesthetics, performance, technical considerations, regulations, and personal convictions. The pursuit of a perfect or final design outcome is subjective, with the focus being on achieving an optimal design instance within the given design requirements and project deadlines. In practice, design is typically divided into manageable tasks among team members. Solutions for these subdomains are iterated upon to generate design variants, resembling a decision tree where some branches remain unexplored while others are pursued or revisited.

2.2 Performance Evaluation of Architectural Design

The AEC industry continuously advances environmentally driven design criteria, including standards like LEED, BREEAM, and DGNB, which promote guidelines for natural resources such as natural daylight. The European EN 17037 guideline serves as a unified and updated European standard, replacing country-specific regulations, and provides standardized strategies for assessing natural daylight qualities in interior spaces. Adoption of this standard improves the evaluation of natural daylight quality by utilizing consistent Lux parameter-based calculations instead of less meaningful glass-to-floor area ratios. This promotes human health, well-being, and performance while also leading to energy savings. In this research, EN 17037 performance-based evaluations are integrated into the design workflow using custom programming and simulation software such as Honeybee & Ladybug (Ladybug 2022) for Grasshopper3D. The inclusion of validation tools in generative programming platforms has made performance-based evaluations more accessible to non-experts, although the presence of multiple and sometimes conflicting performance criteria can pose challenges in evaluations.

2.3 Current Research

The design process seeks to find the best trade-off among often conflicting requirements, whether based on client's brief or architect's convictions. The range of possible solutions is often referred to as "morphospace" (Raup 2006) or "phase space". In this paper, we use the term "solution space" to alignwith the concept of "parametric models" (Woodbury 2010) familiar to architects. Parametric models enable rapid generation of design variations, leading to large solution spaces. Efficient parsing of solution spaces for optimal design solutions is needed. This research presents a workflow combining existing methods to assist in complex design tasks in architectural practice.

Octopus, Opossum, and Wallacei are significant tools for interactive evolutionary computation in architecture. These frameworks, developed by Vierlinger (2013), Wortmann (2017), and Makki et al. (2022) respectively, enhance Grasshopper by offering accessible optimization parameters and interactive features. These user-friendly applications enable multi-objective searches and empower designers with greater control over the numeric optimization process.

Thomas Wortmann addresses the issue of visualizing high-dimensional solution spaces and performance value estimation of unexplored solutions. He surveys various methods of data-visualization and argues that the use of star-coordinates is well suited to display high-dimensional vectors in two dimensions and more intuitive than others, even though it has its own limitations. The two-dimensional representation is overlaid with the estimated performance values to create a performance map (Wortmann 2017).

In a previous work a novel method aimed at increasing this efficacy of solution space navigation through self-organizing maps was explored by Marschall, Aziz and Gengnagel (2016). The approach of self-organized fitness landscapes (SOFL) facilitates the convergence towards optimized design trade-offs between numeric and subjective criteria.

The workflow proposed in this paper builds on these concepts aiming to foster greater accessibility to a larger range of architects by offering an intuitive way of handling high-dimensional solution spaces and performance criteria. While black box optimization techniques are driven by maximizing numerical values and generating only a subsample of the solution space, they tend to make it difficult to integrate design intent and exploration.

3 Methodology

3.1 Introduction

The proposed methodology aims to represent entire solution spaces while incorporating performance estimates for each design option. This is achieved through the development of a Rhino Grasshopper suite called "SearchField". This plug-in enables the generation of discrete solution spaces by systematically exploring all combinations and permutations of input parameters. It also provides dimensionality reduction using an adapted Principal Component Analysis (PCA) to visualize high-dimensional spaces in 2D. By subsampling the solution space, performance simulations can be conducted, and a Linear Regression model estimates performance values for all design options. The methodology includes features for selecting design solutions and filtering the solution space based on design and performance criteria.

"SearchField" is seamlessly integrated with native Grasshopper components and plugins, facilitating a streamlined design workflow that enables designers to generate and evaluate options quickly while considering key performance metrics (Fig. 1).

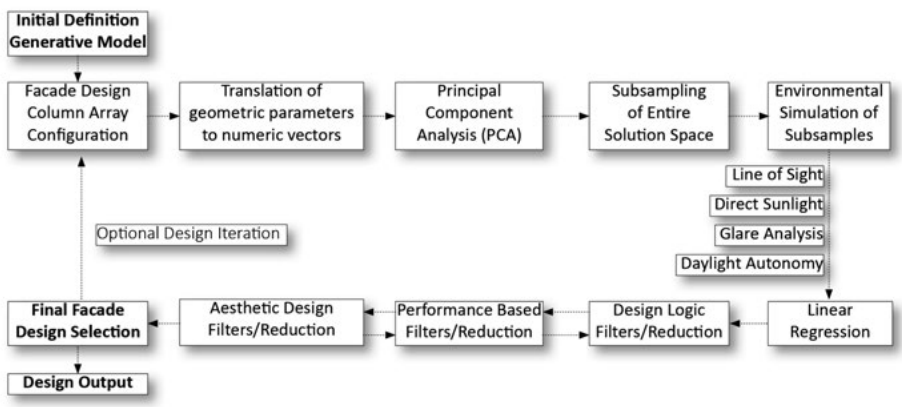

Fig. 1: Process map illustrating workflow using SearchField methodology.

3.2 Case Study

The methodology and custom Grasshopper components were tested on the facade design of an office building currently in planning at the partnering architectural office. The design team aimed to evaluate and refine the basic principle of a glazed facade with a second exoskeleton layer. Irregularly spaced columns supporting the facade would emphasize individual stories and incorporate horizontal overhangs. The investigation focused on the placement of columns for design intent and performance.

The facade consists of 8.1-meter-wide segments, spanning one story in height. Each segment has 16 evenly spaced grid points for potential column placement. To investigate column placement parametrically, this logic is translated into a solution vector. Each dimension or integer in the vector represents the number of grid spacings between columns (Fig. 2b). The vector dimensions range from 1 to 16, with the requirement that the L1 Norm sums to 16. Vectors with lower dimensions are padded with zeros to ensure a consistent dimension of 16 (Fig. 2a). The overall facade appearance is determined by the designer's selection and mapping of segments based on design and performance criteria (Fig. 2c).

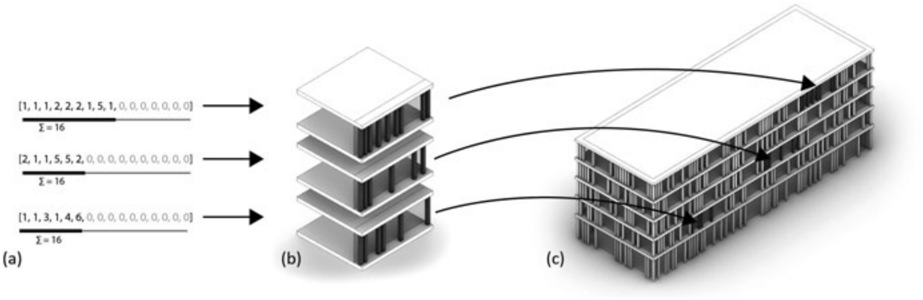

Fig. 2: (a) Solution vectors describing column placement. (b) Geometric interpretation of the vectors. (c) Population of the entire facade.

3.3 Solution space and Dimensionality Reduction

3.3.1 Solution Space

The SearchField components generated all solution vectors, encompassing the entire solution space of 32,768 solutions. While this approach has computational limitations due to combinatorial explosion, it is suitable for architectural design tasks within the capabilities of a standard desktop computer. The generation of the entire solution space took 50 ms, for larger solution spaces, such as 279,936, the time required increases to 500 ms.

This approach offers an advantage over optimization algorithms as it provides the designer with a comprehensive understanding of the solution space's structure. It allows for exploration of the entire range of possibilities without relying on algorithmic guidance.

3.3.2 Dimensionality Reduction

Visualizing high-dimensional data is challenging, but dimensionality reduction techniques, such as those discussed by John Harding (2016), can help by reducing it to two dimensions. Harding's approach, utilizing self-organizing maps, preserves similarity by ensuring that similar solution vectors in high-dimensional space remain close in a lower-dimensional embedding.

To efficiently project high-dimensional vectors to two dimensions, Randomized Principal Component Analysis (PCA) (Martinsson, 2016) using ML.Net was employed (Fig. 3a). Computation time for a dataset of 32,768 objects was 700 ms, and for a dataset of 279,936, it was 2.5 s. The resulting vectors were then transformed by rearranging them on orbits based on their high-dimensional L2 norm while preserving their radial position (Fig. 3b). Vectors with identical L2 norms represent the same input parameter set, albeit in different permutations, making the L2 norm a useful similarity metric in the design process.

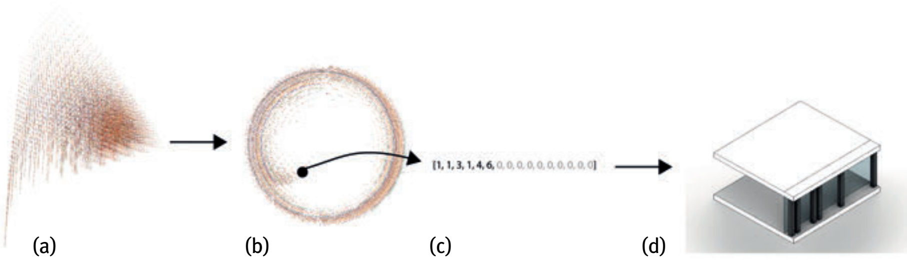

Fig. 3: (a) Projection of solution space to 2D using PCA. (b) Rearrangement of 2D points on orbits using the high-dimensional L2 norm without changing their radial position. (c) Selection of one solution vector. (d) Geometric interpretation.

In the context of star coordinates, Wortmann (2017) addresses the issue of overlapping solutions when vectors with the same L2 norm are projected to the same low-dimensional location. Resolving this, spreading out the two-dimensional vectors on the orbits ensures symmetry and reflects the weightings of the vectors, with larger values at the beginning or end.

Finally, Rhino and Volvox (Zwierzycki et al. 2016) are used to display the two-dimensional solution space as a point cloud, where each two-dimensional vector refers to its high-dimensional counterpart. This representation allows exploration of the high-dimensional space in two dimensions, providing insights into the original data's structure.

3.4 Subsampling, Performance Evaluation and Estimation

3.4.1 Subsampling

The solution space is subsampled to 328 samples, representing 1 % of the total space (Fig.4). Environmental performance evaluation of each sample follows criteria outlined in Sec. 3.4.2, according to DIN EN 17037.

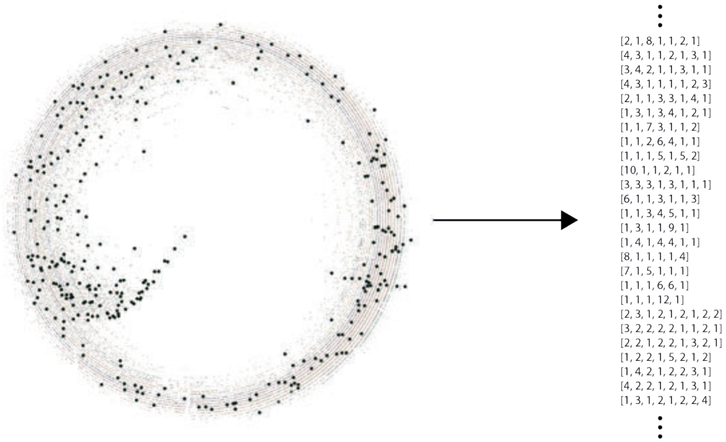

Fig. 4: Subsampled solution space with 328 samples and their vector representation.

To detach performance simulation from the design process, simulations are pre-computed for each sample. This allows integration of various performance criteria and simulation tools, enabling any team member (internal or external) to run the simulation. Results are saved in a structured format and linked to the corresponding solution vector.

3.4.2 Performance Evaluation

Our study, based on DIN EN 17037 “Tageslicht in Gebäuden”, aimed to conduct four evaluations to develop an optimum facade design selection strategy: spatial daylight autonomy (sDA), line of sight to the outside (LOS), direct sunlight (DS) on the facade, and glare analysis (GA). Ladybug and Honeybee plug-ins accessed through Grasshopper were used for sDA and glare analysis, while custom programming in C# within Grasshopper was developed for LOS and DS analysis.

It’s worth noting that each analysis requires varying levels of detail (LoD), regarding the geometric input and environmental meta information (e. g., material attributes for the sDA analysis), and simulation runtimes, ranging from seconds to hours or days.

Each of the 328 subsampled solutions underwent performance evaluation using an $8.1 \times 6m^2$ shoebox geometry for all four cardinal directions.

Line of sight

In the initial design iteration, a subset of column configurations was assessed for visibility to the building's surroundings. This assessment considered three factors: horizontal viewing angle,distance to the outside perimeter, and visibility of at least 75 % of the room surface. A custom script was developed to generate a multidimensional score for each criterion, ranging from 3 (lowest score) to 9 (highest score), see Fig. 5. Using this script, all 328 design instances were quickly evaluated.

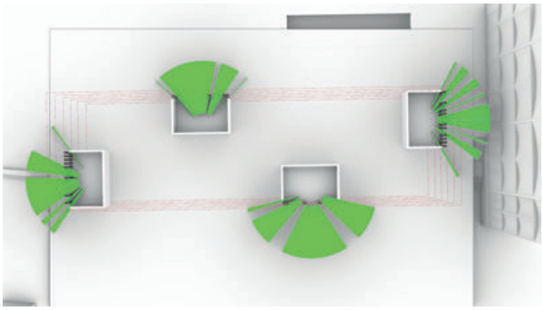

Fig. 5: LOS analysis, exemplary simulation sequence.

Direct Sunlight

Direct Sunlight (DS) analysis evaluates natural sunlight quality in buildings. It commonly uses the equinox date of March 21st as a reference. The duration of direct sunlight exposure is determined by the allowed sun elevation angle, e. g., 11° for Germany (Fig. 6). Boundary conditions such as climate zone, facade geometry, building construction, and obstructions affect sunlight accessibility. Results below or equal to 1.5 hours are considered poor, while 4 hours or more indicate good and well-lit spaces.

Glare Analysis

The glare analysis utilized the EnergyPlus Weather (EPW) format for Berlin and the Ladybug and Honeybee plug-in suite in Grasshopper. The Evalglare method calculated the Daylight Glare Probability (DGP) based on user assessments, categorizing discomfort levels. A high dynamic range imagery (HDRI) technique was employed using a 180° fish-eye camera lens to capture source-target images and generate color-coded False Color representations (Fig. 7). The analysis involved cycling through subsamples for each cardinal direction, exporting resulting images and DGP values.

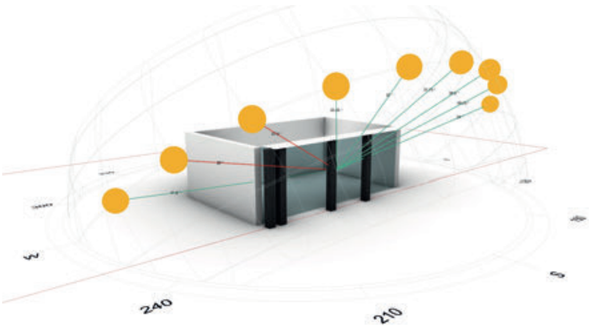

Fig. 6: DS Analysis, exemplary simulation sequence.

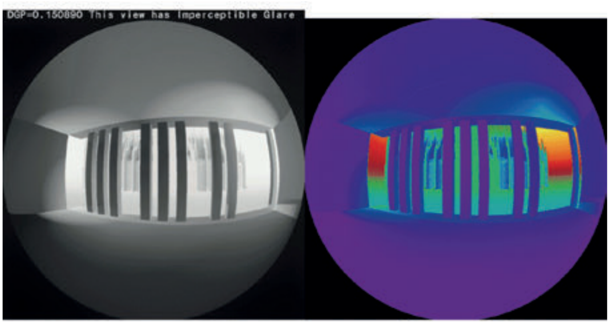

Fig. 7: Glare Analysis, exemplary simulation sequence.

Spatial Daylight Autonomy (sDA)

Spatial Daylight Autonomy (sDA) was used to assess daylight provision in this study, a widely used performance metric in building design and analysis (Nabil and Mardaljevic 2005). sDA quantifies the spatial and temporal distribution of daylight in interior spaces by measuring the percentage of occupied hours per year where a space exceeds a predefined target illuminance level. Factors such as window positioning, building orientation, shading devices, and surface reflectivity are considered in calculating the sDA value. Higher sDA values indicate better illumination throughout the year, improving occupant comfort, health, and productivity. The simulation process automated performance evaluation for each variant's cardinal directions, streamlining the analysis. Specific target values, such as a minimum of 300 lux for 50 % of daylight hours in 50 % of the area for a "poorly lit space", are used to determine adequate daylight levels. A high score is achieved when spaces achieve equal to or greater than 750 lux for 50 % of the area and daylight hours (Fig. 10).

Tab. 1: Recommendations for the supply of daylight through dayligth openings in vertical and inclined surfaces.

Level	E_T	$F_{plane,\%}$	min E_T	$F_{plane,\%}$	$F_{time,\%}$
low (score 1)	300 lx	50 %	100 lx	95 %	50 %
middle (score 2)	500 lx	50 %	300 lx	95 %	50 %
high (score 3)	750 lx	50 %	500 lx	95 %	50 %

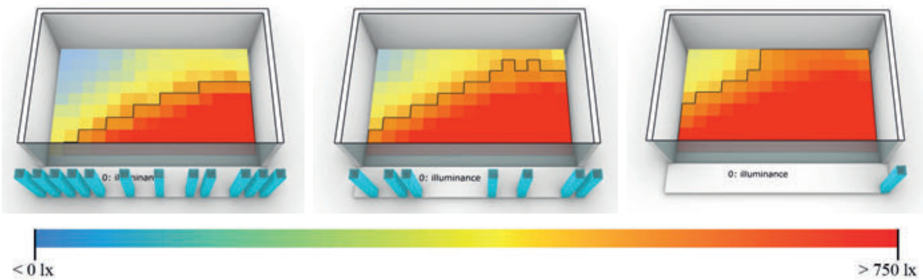

Fig. 8: Exemplary visualization of achievable scores applying DIN EN 17037 guidelines. Left: Sample 20 / West, Score 1; middle: Sample 200 / West, Score 2; right: Sample 332 / West, Score 3.

3.4.3 Performance Estimation

After performing the evaluation of the samples, Linear Regression Models were trained for each performance criteria to estimate the performance of the remaining 32,440 solutions. The chosen Machine Learning model was Linear Regression with a trainer based on the Stochastic Dual Coordinate Ascent (SDCA) method, known for optimizing convex objective functions (Yu et al. 2011). The trained Linear Regression Model was applied to the high-dimensional vectors, and Fig. 9 demonstrates its reliable performance prediction.

Using color-coding, a heat-map can generate a visual representation of performance values in the 2D solution space. This enables an intuitive identification of well-performing directions, aiding in the quick identification of the most favorable solutions.

Tab. 2: R-Squared (R2): Represents predictive power of model, the closer to 1.0 the better Mean absolute error (MAE): measures how close the predictions are to the actual outcomes, the closer to 0.0 the better Root Mean Squared Error (RMSE): measure of accuracy, the closer to 0.0 the better.

	LOS	DSs	DSw	DSn	DSe	GAs	GAw	GAn	GAe	sDAs	sDAw	sDAn	sDAe
R2	0.81	0.83	0.83	0.83	0.83	0.82	0.76	0.69	0.74	0.55	0.61	0.60	0.59
MAE	0.04	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.08	0.09	0.08	0.08
RMSE	0.06	0.02	0.02	0.02	0.02	0.01	0.01	0.02	0.02	0.11	0.12	0.11	0.11

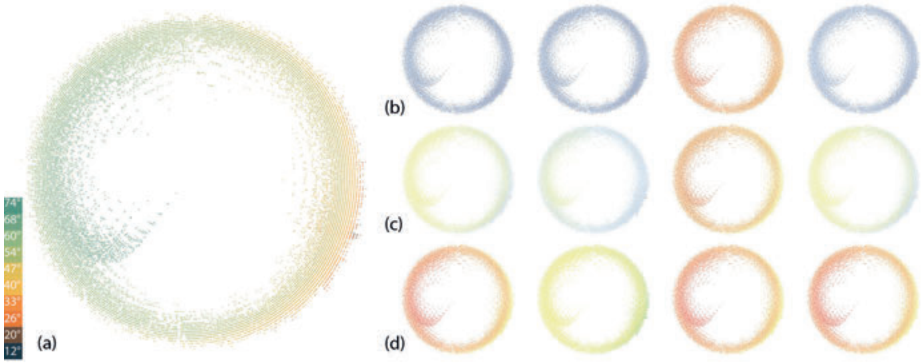

Fig. 9: Performance estimates: (a) Horizontal view angle; (b) sDA east, north, south, west; (c) Glare east, north, south, west; (d) DS east, north, south, west.

3.5 Design Process

3.5.1 Interaction, Selection and Filtering

To efficiently display the solution space and select options, two data structures are utilized: a Rhino point cloud and a KD-Tree (Panigrahy 2008). By specifying numerical ranges for performance criteria, the solution space can be filtered and reduced, allowing for a quick evaluation of design directions that meet performance requirements. Design filters, defined using regular expressions (.NET Regular Expressions 2022), are applied to the high-dimensional solution vectors, which describe design criteria through their properties. The actual design geometry is generated and displayed upon selecting solution vectors. The SearchField components currently offer two selection methods: around a point with a radius and a slice (Fig. 10).

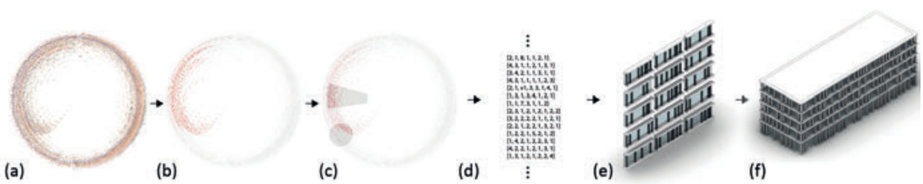

Fig. 10: (a) Projected solution space; (b) Filtered solution space; (c) Selection of design variants; (d) High-dimensional solution vectors of selection; (e) Design specific interpretation of vectors; (f) Population of facade using segments.

3.5.2 Iterative Design Process

The designer can explore and map selected solutions to the building (Fig. 10a), gaining insights into design variations and their performance properties. Filters can be defined to exclude solutions that don't meet specific criteria. For example, to filter out all facade segments that have two columns directly nextto each other, all vectors containing the integer 1 would be excluded, resulting in 610 possible design options for the facade segments. Alternatively, filters can be applied to include solutions within a desired performance range. The designer can freely browse options and gradually filter out undesired solutions, resulting in a refined and traceable design.

In the initial design process, criteria were defined to allow a maximum of 6 adjacent columns with a spacing of up to 10 grid units. Performance filters were then applied to include only options in the upper third of south-facing performance values, resulting in 1906 design options (Fig. 11). In the final step, stricter design criteria were implemented, limiting adjacent columns to a maximum of 3 and maintaining a maximum spacing of 10 units. Performance demands were adjusted accordingly, with different criteria for each cardinal direction. For the south-facing direction, the mid-third of performance values was used, resulting in 441 options. The east-facing and west-facing directions considered the upper third for Glare Analysis and the mid-third for Direct Sunlight, Line of Sight, and spatial Daylight Autonomy, yielding 474 options. The north-facing direction allowed for all upper thirds, resulting in 75 options. This final design outcome represents an optimal solution achieved by incorporating design and multiple performance criteria within these boundary conditions.

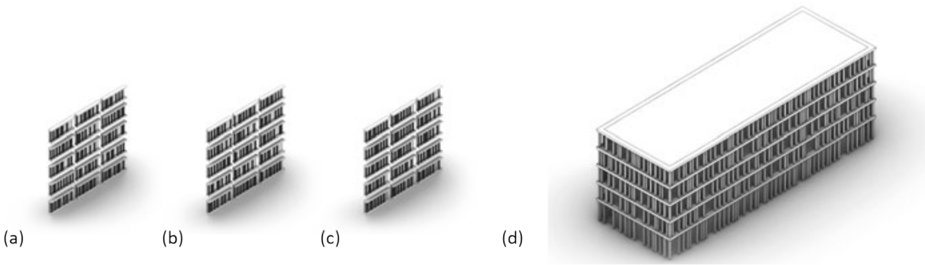

Fig. 11: (a) South-facing facade elements 15/441. (b) East- and West-facing facade elements 15/474. (c) North-facing facade elements 15/75. (d) Final design of column placement.

4 Conclusion and Future Work

Our approach enables intuitive navigation of high-dimensional solution spaces, integrating performance criteria and domain expertise without requiring deep tool understanding. It improves architectural design outcomes by incorporating performance evaluation in the early stages. Currently, we focus on spaces up to 5 million solutions, using dimensionality reduction and machine learning for fast computation. Future steps include expanding to larger spaces, employing advanced techniques like Manifold Learning for projection and neural networks for value estimation. We aim for a step-by-step design process that narrows the solution space and refines performance metrics.

Future research should focus on improving collaboration between architectural designers and domain experts, refining and exploring the validation process and analysis environment for performative evaluations. A standardized and robust digital evaluation environment is crucial to reduce human errors in geometry and numeric input declaration for simulations.

We developed an interactive interface for designers to navigate and preselect design variations based on different criteria. To understand the high-dimensional solution space, we implemented a visualizationassistance using arrayed concentric 2D point-clouds. Using 3D surrogate geometries, designers can allocate cluster characteristics on the 2D map. Further research and empirical testing are needed to create a user-friendly interface and compare it with other computational design strategies. Our goal is to demonstrate that SearchField enables informed decision-making and exploration of a significantly larger solution space.

References

Carpo, M. 2017. *The Second Digital Turn: Design Beyond Intelligence.* The MIT Press. DOI: https://doi.org/10.2307/j.ctt1w0db6f.

Harding, Jo. 2016. Dimensionality Reduction for Parametric Design Exploration. *Advances in Architectural Geometry.* Eds. M. Kohler et al. Vdf Hochschulverlag ETH Zurich.

Hamedani, A. and F. Huber. 2012. A comparative study of "DGNB", "LEED" and "BREEAM" certificate system in urban sustainability. DOI: 0.13140/2.1.3177.0568.

Kohler, N. and S. Moffatt. 2003. Life-cycle analysis of the built environment. *Industry and environment* 26.2. 17–21.

Makki, M., M. Showkatbakhsh, Y. Song. 2019. Wallacei Primer 2.0. Available online: https://www.wallacei.com/. Accessed on 12 December 2022.

Marschall, M., S. Aziz, and C. Gengnagel. (2016). Alternative means of navigating parameter spaces.

McNeel, R. 2010. Rhinoceros 3D, Version 6.0. Robert McNeel & Associates. Seattle. WA.

Martinsson, P.-G.. 2016. Randomized methods for matrix computations and analysis of high dimensional data. DOI: 10.48550/arXiv.1607.01649.

Nabil, A. and J. Mardaljevic. 2006. Useful daylight illuminances: A replacement for daylight factors. *Energy and Buildings* 38, 905–13. DOI: 10.1016/j.enbuild.2006.03.013.

.NET Regular Expressions. Microsoft Docs, 15 September 2021, https://docs.microsoft.com/en-us/dotnet/standard/base-types/regular-expressions. Accessed 5 August 2022

Panigrahy, R. 2008. An Improved Algorithm Finding Nearest Neighbor Using Kd-trees. In: Laber, E. S., Bornstein, C., Nogueira, L. T., Faria, L. (eds) LATIN 2008: Theoretical Informatics. LATIN 2008. Lecture Notes in Computer Science, vol 4957. Springer, Berlin, Heidelberg.

Raup, D. 2006. The Geometry of Evolution Adaptive Landscapes and Theoretical Morphospaces. pp. 57–70. Cambridge University Press. DOI: 10.1017/CBO9780511618369.005.

Rittel, H. W. J., and M. M. Webber. 1973. Dilemmas in a General Theory of Planning. *Policy Sciences,* 4(2), 155–69.

Software Addon Ladybug and Honeybee https://www.ladybug.tools/

Suk, J. Y. and M. Schiler. 2012. Investigation of Evalglare Software, Daylight Glare Probability and High Dynamic Range Imaging for Daylight Glare Analysis. *Lighting Research and Technology.* DOI: 10.1177/1477153512458671.