SYMPOSIUM ON SIMULATION IN ARCHITECTURE + URBAN DESIGN
SimAUD 2020

The Development of Optimization Methods in Generative Urban Design: A Review

Yufan Miao[1], Reinhard Koenig[2] and Katja Knecht[3]

[1]Future Cities Laboratory, Singapore, Singapore, miao@arch.ethz.ch
[2]Bauhaus University Weimar, Weimar, Germany, reinhard.koenig@uni-weimar.de
[3]Future Cities Laboratory, Singapore, Singapore, katja.knecht@arch.ethz.ch

ABSTRACT

Design optimization, as one of the major generative design methods, has been studied ever since the 1960s. However, half a century later, in an urban design context, it is still in the research and experimental phase and rarely employed in actual projects. This paper aims to reflect design optimisation methods through a comprehensive review from the perspective of historical development, competitors' success and future trends. It also proposes a conceptual framework to enhance current optimisation methods with machine learning for generative urban design.

Author Keywords

Design Optimisation, Machine Learning, Urban Design

1 INTRODUCTION

Traditional design methodologies are based on the separation between analysis and synthesis [22]. In addition to the traditonal methods, Computer Aided Architectural Design (CAAD) systems have emerged as a design support tool in the 1970s [36], inheriting from Computer Aided Design (CAD) the capabilities for "creation, modification, analysis, or optimization of a design" [30]. To bridge the gap between analysis and synthesis, modern CAAD tools are expected to provide an integrated, domain-oriented and knowledge-based environment. When the criteria chosen for design can be quantified and expressed mathematically, the design problem can be naturally formulated as an optimization problem [5], which provides an avenue to integrate design analysis with synthesis.

Optimization, as one type of generative design methods, has been applied to architecture since 1960s [26]. However, half a century later, in an urban design context, it is still in the research and experimental phase and rarely integrated into mainstream CAD software [73]. In contrast to design optimization, rule-based modeling as a generative method achieved a greater application in practice through the success of the 3D modeling software CityEngine [20], which dominates the commercial market especially after it having been acquired by ESRI [19] and having been fully integrated into their Geographic Information System (GIS) products. CityEngine's success originated from its strong theoretical background of shape grammar [69] and L-system [59]. In comparison to rule-based methods, design optimization has such advantages as the integration of design analysis and generation, a large solution space, and disruptive innovation [67], which are necessary for architectural and urban design problems. However, the lack of mathematical formulations for design problems and the issues of inadequate efficiency and effectiveness are shortcomings of design optimization based generative methods [44]. Therefore, a hybrid approach with machine learning could be proposed and studied to take advantage of both.

Recently, with the victory of Google's AlphaGo [28] over human professionals in the Go game, machine learning has ignited people's passion for Artificial Intelligence (AI). New generative methods such as Generative Adversarial Network (GAN) [29], and Variational Auto-Encoder (VAE) [17] have achieved great success in fields such as computer vision. However, such methods have rarely been applied to urban design problems. A remarkable exception in this aspect is the StreetGAN model [31], which applied GAN model for the generation of street networks. However, design results from machine learning are difficult to be interpreted whereas interpretability is of vital importance during the design discussion between designers and stakeholders. Although interpretable machine learning emerged as a hot topic recently [57], it is still at the early stage. In comparison with machine learning methods, design optimization results are easier to be interpreted by user-specified objective functions. However, it is difficult to accumulate design knowledge to improve the search path in future optimization. Therefore, a combination of the latest machine learning methods and optimization methods are expected to complement each other.

Due to the complexity of urban design problems themselves, Evolutionary Multi-Objective Optimization (EMO) methods are usually employed [53]. However, the performance in terms of speed and consistency is often criticized [78]. For architectural design, machine learning based surrogate models for single objective optimization problems have been proposed as an alternative [76]. Nevertheless, in urban design, multi-objective optimization problems are more popular due

SimAUD 2020 May 25-27, Online

to its multi-objective nature [7]. Therefore, to improve the performance, surrogate models could be introduced to partly replace the computationally intensive part of the EMO process.

Latest review papers are often written from the perspective of the authors' background and specific to one aspect of urban design [63, 21, 44]. This paper, on the contrast, aims to provide a comprehensive review from the perspective of the historical development, the current competitors, and the future trend of optimization methods in urban design. Through the review and reflections, a conceptual framework will be proposed to enhance design optimization with machine learning.

In accordance, the following questions are supposed to be answered by the paper:

- What are the advantages and challenges of design optimization in urban design?
- What could design optimization learn from the success of other methods such as shape grammar?
- How can the optimiztion methods be combined with the latest machine learning and surrogate methods to take advantages of both?

The remainder of the paper is divided into four sections, namely, the historical pathway, the peer pressure, the future trend, and finally the conclusion and outlook. In the historical pathway section, the development of design optimization in architectural and urban design is reviewed in chronological order. The major advantages and challenge questions are answered in this section. In the peer pressure section, a comparison between design optimization and other popular generative design methods is presented. What can be learnt for design optimization from other methods is discussed in this section. In the future trends section, surrogate methods and machine learning based generative methods are reviewed. A conceptual framework to enhance optimization techniques with machine learning is proposed. In the last section, the paper is summarized and an outlook on possible future work is provided.

2 THE HISTORICAL PATHWAY

As Confucius said, "Study the past, if you would define the future". An overview of the historical development (table 1) could provide us with a clear picture of the evolution of design optimization methods in architectural and urban design. From this picture, it is expected that the advantages (table 2) and challenges (table 3) of the methods can be better understood from the historical perspective. It is also interesting to see how design optimization methods and techniques evolved to provide the latest cutting edge technology to meet the constantly changing needs in architecture and urban design. The first question is to be answered in this section.

Design optimization started firstly in architectural design before it was scaled up to urban design. In academia, attempts to apply optimization methods to design problems can date back to as early as 1969 from Simon's remarkable paper the "Science of Design" in his seminal book "The Sciences of the Artificial" [65]. During this phase, there was even no well established theory for architecture yet [75] and design optimization was motivated to be one of the numerical attempts to prove the scientificity of architecture besides its inborn artistic nature. Simon [64] also re-defined design creativity as the richness of a combinatorial space that architects move through and add one element after another to, which coincides exactly with mathematical optimization. Gero [27] reviewed a range of applications of optimization techniques in architecture and urban design. He pointed out that the lack of numeracy in architectural education and the lack of numerical models in architecture hindered the applicability of this method in design. Mitchell et al. [51] also contributed by proposing an optimization method to synthesize small rectangular floorplans. Endeavors continued for decades to develop optimization-based methods for design in the domain of CAAD. In 1980s, Gero, Radford and Balachandran [25, 2, 60] introduced multi-objective optimization in design through several publications of theirs. During this phase, design optimization did solve a few sub-problems of architectural design. However, the employed numerical optimization methods often failed when the design problems could not be mathematically formulated.

In 1990s, more logic-based AI techniques were introduced to weaken the conditions of mathematical formulation while debates about the necessity of such methods for design was heated. Pohl et al. [58] proposed a prototype of an intelligent computer-aided design system that emphasized on the partnership between computer and human. Schmitt [62] pointed out that the research frontiers of CAAD started to shift from design automation to design support and appealed for more insights from human cognition. Malkawi [46] proposed a design-oriented method to evaluate, critique and optimize energy use and design in buildings. Many efforts were made in this decade, but doubts and critiques were ubiquitous from outside the CAAD field.

Flemming [23] strongly defended the field by pointing out the four common fallacies in the critiques. They are 1) treating design as a monolithic and indivisible process, whereas design is iterative and consists of a multitude of subtasks; 2) expecting that architectural practice should be supported as is without challenging the status quo whereas current practice is not without shortcomings, which the design and development processes of CAAD can help identify and address; 3) CAAD being just an application of linguistic analogy to design; and 4) traditional design being a top-down approach of critique that relies on authorities and their theories, in particular Heidegger's, to discredit CAAD systems and approaches thereby disregarding or neglecting empirical evidence from the field, users' needs and experiences.

However, what could not be denied was that the applicability of the developed systems were still very limited. Meanwhile, critical views were also voiced internally from the field and motivated the research agenda to move forward. A famous example is the CAAD's seven deadly sins brought up by Maver [48], including macro-myopia, déjà vu, xenophilia, unsustainability, failure to validate, failure to evaluate, and failure to criticize. Motivated by both external and internal critiques, new methodologies started to emerge such as using genetic programming to explore design spaces [6].

At the beginning of the 21st century, benefiting from the burgeoning computing power, more derivative-free and stochastic optimization methods were realized and applied to complex discrete nonlinear problems. Coates et al. [13] initiated several projects in his Centre for Environment and Computing in Architecture (CECA), including the use of generative algorithms to construct forms for architectural design. Michalek et al. [49] proposed an optimization method for floorplan layout design with both simulated annealing and genetic algorithm. They also proposed an interactive method for architectural layout optimization [50]. Wright et al. [79] tried to optimize building thermal design with the help of a multi-criteria genetic algorithm. Caldas & Norford [8] proposed a design optimization tool based on a genetic algorithm to optimize the environmental performance of buildings. Wetter & Wright [74] made a comparison between deterministic and probabilistic optimization algorithms for non-smooth simulation-based optimization. They drew the conclusion that stochastic or hybrid methods such as particle swarm with Hooke-Jeeves algorithm were the best choices in practice when gradient-based methods failed because of large discontinuities in the cost functions.

For urban design, Derix [16] used Ant Colony Optimization to generate street networks and Quantum Annealing to find out desired adjacencies among different land use units. Moreover, new insights about the usage of optimization methods emerged. Bleiberg & Shaviv [3] used optimization to enhance collaborative design. At the same time, researches in Multi-Objective Optimization also had breakthroughs with remarkable algorithms developed such as SPEA2 [80], NSGA-II [15], and later HypE [1]. However, although more and more physical realities were achieved with the help of CAAD, such as the Science City Zurich, design support tools failed to make significant inroad into design practice [61].

In the past decade, research progress continued in design optimization within the field of CAAD, extending from architectural design to urban design. The project KAISERSROT is a successful example where CAAD was massively used in architectural and urban design practice [68]. In this field, many researches were focused on design space exploration. For design exploration, Janssen [35] proposed an evolutionary system. Turrin et al [71] developed a method with combined parametric modelling and genetic algorithms for design exploration of performance driven geometries. Stouffs [70] also proposed methods to combine generative with evolutionary exploration. Meanwhile, model-based optimization has been proved to be a more efficient alternative to evolutionary algorithms in terms of speed and feasibility [77] when the problem is or can be re-formulated as a single objective optimization problem. For Multi-Objective Optimization problems, hybrid methods with both metaheuristic and model-based optimization would be ideal as was already proved in other engineering design fields [66].

Objects	Developments
1960s-1980s	early ontological attemps; single objective optimization
1990s	methods further developed; logic-based, human-centered methods
2000s	more complex nonlinear problems; meta-heuristic optimization methods; multi-objective optimization methods; simulation-based methods
2010s	data-driven AI methods

Table 1. The historical development of the design optimisation methods.

	Advantages
1960s-1980s	solve sub-problems of architectural design; prove the scientificity of architecture
1990s	attracted more attentions from design field; new methods attempted at the application level
2000s	computing power for more complex problems; applications emerged for urban design
2010s	AI technologies augment human creativities; availability of large amount of data; more statistical models; successful application to design practice

Table 2. Advantages of the optimisation methods in different ages

	Challenges
1960s-1980s	lack of well defined mathematical models
1990s	debates from both the internall and external of the CAAD community
2000s	theoretically thriving but practically immature
2010s	mismatch between the data-driven approaches and the design logic and philosophy

Table 3. Challenges of the optimisation methods in different ages

With the development of spatial analysis techniques such as space syntax [33], an increasing number of quantitative evaluatin methods have been introduced to urban design, which enrich the design requirements design optimization could fulfill. Celani et al. [11] proposed a method to generate urban patterns by combining shape grammars and genetic algorithms. Cao et al [9] used Multi-Objective Optimization methods for land use planning. Koenig [39, 40] proposed a synthesis method for street network and building layouts based on EMO. Moreover, Koenig & Schmitt [41] also proposed the Cognitive Design Computing (CoDeC) framework. Different from earlier design optimization attempts, their methods not only aim at synthesizing urban geometries but also target at tackling a major critique of computational creativity, which is the lack of humanity [14] by reinforcing the role of human intelligence through designers' interactions with the generated urban design. The core of their approach is the EMO-based generative methods which could bring to urban design benefits such as transparency [52] and integrativeness [67]. Moreover, this research aims to address one of the major challenges in EMO for generative urban design, namely the representation problem. Their recent work [37] has successfully applied EMO to the generation of multiple urban design layouts with urban objects such as street network, blocks, parcels and buildings. Despite successful work crossing research frontiers, in practice, there is still a long path to go. The lack of quantified design evaluation measures and metrics is still a big challenge. Moreover, computer generated design solutions are usually simple and only suitable for prototyping. Due to the nature of EMO, highly intensive computation is usually required, even for simple generation, which lags behind real-time. These problems are expeced to be solved with a hybrid approach. The following section will feature a comparison between design optimization and other existing generative design methods.

3 THE PEER PRESSURE

As Confucius said, "When I walk along with two others, they may serve me as my teachers". As one kind of generative design method, design optimization has a great potential to form the basis of future AI-based integrated design support tools. For the time being, it still has a lot to improve to meet the expectations, which can be learnt from its peer generative methods. Generative design methods have already been comprehensively reviewed by Singh and Gu [67]. In their review paper, they identified five major types of generative methods, namely, Cellular Automata, Shape Grammars, L-Systems, Swarm Intelligence, and Genetic Algorithms. This paper was written over 7 years ago and thus does not cover the latest technologies such as machine learning based methods. This section aims to compare existing generative design methods and provide some insights about what can be learnt for design optimization from other generative methods.

Cellular Automata has been successfully applied to architectural and urban design in many aspects [32, 42]. However, due to its bottom-up nature, the outcomes are often complex and difficult to predict. Moreover, to define rules, which are required to guide the generative process, is also unfamiliar to design practitioners. In recent research publications, interests in this method have already faded. On another hand, Swarm Intelligence has also been applied to architectural and urban design applications. The popularly used Agent-based model (ABM) embodies the essence of this method. One famous example is the MATSim [34] as a successful ABM system for transportation planning. However, these kind of models are more suitable for social and collective behavior studies [10, 45]. This can be employed as one dimension of the urban design task but the whole design should go beyond that. For design optimization, agent-based models can be used as for design evaluation and the visualisation of the simulations could help a designer to better understand the future behaviors and movements of people in the designed space.

Both Shape Grammar as well as L-System constitute the so-called rule-based procedural modeling techniques [56], which have been applied for the generation of both architectural and urban design. From shape grammar, two major branches have emerged, namely, Grammars for Designing (GfD), and Grammars Derived from Designs (GDfD) [47].

GDfD is usually achieved by evaluating the rules and shapes derived from the analysis and then selecting the suitable ones for the generation of a new design. Successful studies have been conducted by Duarte et al [18] and Paio et al [55]. Both of them tried to introduce optimization into the generative process as well, which is a way to combine design optimization and procedural models.

However, to date GfD have achieved the greatest level of success and adoption in practical architectural and urban design. A striking example is the 3D modeling software CityEngine [20]. Due to its integration with GIS products, CityEngine provides a convenient platform for designers by combining location-based data with capabilities for design generation allowing buildings to be modeled with high visual quality and level of detail. CityEngine originally evolved from the first school with the shape grammar of Computer-Generated Architecture (CGA). The limited capabilities of complex shape grammars to support user interaction have in a recent version of CityEngine been solved by introducing interactive design capabilities [20]. However, there is still a drawback that originates from the limitation of GfD approaches in general. The generated designs are usually independent from the urban context rather than from existing designs. They are more suitable for use in a large variety of contexts rather than those with strong local characters. The success story of CityEngine, reasons for its adoption in design practice but also its remaining shortcomings, can help inform future AI-based integrated design support tools.

In comparison to shape grammar, design optimization methods themselves involve highly complex and computationally intensive processes and thus are not ideal for fast and large scale design generation with limited computing power. The advantage design optimization has is the so-called intelligence as a mimic of the human design process. However, the current state of such intelligence is still primitive. Moreover, it is still debatable whether such intelligence is necessary and how the roles human and computer should be defined respectively in the design process. Shape grammar based approaches, on another hand, focus on the generation of large amount of complex 3D geometries which provide users a comprehensive visual thinking environment without interfering with the actual design process. Although this is not aligned with the current heat waves of AI, its success has proved that it is currently the best engineered solutions for design. Design optimization should at least catch up with the speed and scale of shape grammar based software such as CityEngine to be superior. This is also why hybrid methods with machine learning are necessary and will be explained in the following section.

4 THE FUTURE TRENDS

With the success of AI researches, machine learning based approaches are drawing increasing interest from academia. For generative design, these methods are applied in general in two ways. The first way is through model-based optimization whereas the second is through generative models.

For model-based optimization, researchers use machine learning methods to construct the surrogate models to approximate the behaviors of simulations. Remarkable researches in this aspect are from Wortmann [76, 77, 78] where surrogate model based methods were applied to architectural design. Based on his publications, this kind of methods are superior because of the reliable convergence rate and relatively fast

Objects	Advantages
CityEngine	fast generation speed; large scale; GIS data interoperability; 3D visualization and interaction
Design Optimisation	integration of analysis and generation; large number of design alternatives; the emergence of novel solutions

Table 4. A comparison of the advantages of CityEngine and Design Optimisation.

	Disadvantages
CityEngine	separation of analysis and generation; separation of human and computer; difficult to generate novel solutions
Design Optimisation	slow; complex; limited scale; rudimentary results

Table 5. A comparison of the disadvantages of CityEngine and Design Optimisation.

speed. It is also a method popularly employed in energy-driven architectural and urban design [54, 63] where expensive simulations such as computational fluid dynamic simulation is usually popular. This kind of methods are usually applied to single objective optimization problems. For multi-objective optimization problems, different objectives are usually mathematically combined and transformed into a single objective problem. The transformation itself is a difficult task and usually relies on expert knowledge. For urban design, since the objectives can be of a large number, the transformation of multi-objective problems into single objective can be tricky especially when they are correlated to each other.

For generative models, the most popular ones are GAN and VAE. They have achieved great success in the research field of Computer Vision (CV). Therefore, most of the variants of these two methods are designed for image data. GAN model is based on the philosophy of zero-sum game [43], which contains a generator and a discriminator. The former is responsible for generating data with the model learnt from data whereas the latter tells whether the generated data is close to the real data. The model is optimized continuously until the discriminator can hardly tell the difference between generated data and real data. For design generation, streetGAN as a variant of GAN has been successful [31]. This kind of researches have been seldomly employed in urban design but exhibited potential to enhance the current design optimization methods. For single urban geometries such as street networks, streetGAN has already provided a successful example. Similar approaches could be possibly applied to blocks, parcels and buildings. What would be interesting is how the machine learning method could capture the inter-relationship between different types of urban geometries such as that of street network and parcels. VAE, on another hand, adopted a different philosophy. It is to reduce the dimension of the original data and represent the data as an encoder with reduced but key features. Then it uses a decoder to re-generate the original data space. The encoder is optimized through back propagation of the error between original data and generated data. In this process, novel data can also be generated [4], which is important for design. A similar approach that reduces the dimension of data and only depicts key features is the Self-Organizing Map (SOM) which has been proved to be useful for architectural and urban design [24].

Concepts about employing machine learning methods for a recommendation system for urban design were also proposed by Chirkin and Koenig [12]. In their framework, unsupervised learning methods were used to create design quality measures extracted from user preferred designs. These measures were later employed to improve the design draft through optimisation. This research inspired the proposal of a new framework as in figure 1. Similarly, the purpose is to enhance optimization methods with a learning capability. However, different from Chirkin and Koenig's approach, this new framework not only tries to provide alternative methods to optimisation for design generation when different user specifications are input but also tries to avoid excessive usage of optimisation methods when unnecessary. Moreover, as has been found by von Buelow [72], there is a selection problem for the users to choose the right option from massive generated design variants. Also, a performance optimized solutions might not be aesthetically pleasant. For the first problem, SOM as a machine learning method, has been employed as an attempt to reduce the solution space [38]. For the second problem, solutions such as Opossum developed by Wartmann [77] were proposed to provide the designers a visualisation interface to compare the geometries and the Key Performance Indices (KPI) of both manual designs and generated designs. For both problems, machine learning methods are potentially helpful to learn from the users' selection as feedbacks to guide the future generative process.

There are five ways of inputs expected from the users. They are quantitative design objectives, parametric design specifications, geometries, maps or design proposals in image format, and semantic specifications. Quantitative design objectives are well-formulated objective functions and constraints that can be directly input into the optimization process to guide the design generation which is the traditional way of implementing design optimization. The parametric design specifications are expected to be used by the rule-based generators for fast and large scale urban geometry generation. The generated geometry can either be outputed directly or improved further by EMO with specified objective functions after being converted into a genotype in the data structure encoder. The framework should also be able to accept already existing designs in the format of either geometries or images. The geometries can be decoded to retrieve similar genotypes for evolutionary algorithm based geometry generation or similar statistical models for machine learning based image generation. The data assignment and retrieval tasks are handled by a database management system. Moreover, the generated design solutions can be manually reviewed and selected by designers with the selected ones being stored in the database.

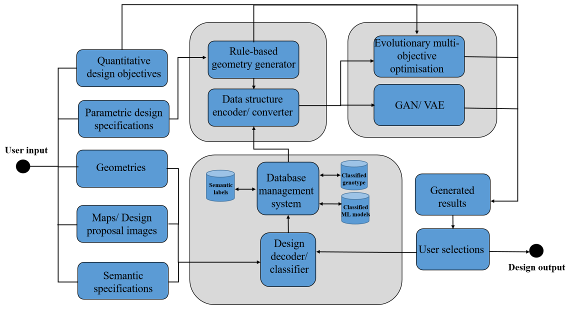

Figure 1. A conceptual framework to integrate design optimization with machine learning (the groups in grey color are expected to run in the background).

5 CONCLUSION AND OUTLOOK

In this paper, key advantages and challenges for design optimization in architectural and urban design have been reviewed from a historical perspective. Moreover, as a generative design method, design optimization and other kinds of generative design methods are compared. What design optimization could learn from other methods is also proposed. Furthermore, the shortcoming of the optimization methods could be enhanced by the latest data-driven approaches such as machine learning. How this could be achieved is explained in the proposal of a conceptual framework.

As this paper constitutes a literature review of different methods and is therefore set at a conceptual level, a concrete description of a possible implementation of the framework is beyond its scope. The implementation of the proposed framework poses several challenges, such as how the data structure encoder be implemented in the proposed framework and how the hybrid database should be designed for the storage of different types of data. These questions need to be answered in the future research and the viability of the conceptual framework also needs to be affirmed by actual implementations.

REFERENCES

1. Bader, J., and Zitzler, E. Hype: An algorithm for fast hypervolume-based many-objective optimization. *Evolutionary computation 19*, 1 (2011), 45–76.
2. Balachandran, M., and Gero, J. A knowledge-based approach to mathematical design modeling and optimization. *Engineering Optimization+ A35 12*, 2 (1987), 91–115.
3. Bleiberg, T., and Shaviv, E. Optimisation for enhancing collaborative design. In *Proceedings of the Building Simulation* (2007).
4. Bozkurt, A., Esmaeili, B., Brooks, D. H., Dy, J. G., and van de Meent, J.-W. Can vaes generate novel examples? *arXiv preprint arXiv:1812.09624* (2018).
5. Braibant, V., and Sander, G. Optimization techniques: synthesis of design and analysis. *Finite elements in analysis and design 3*, 1 (1987), 57–78.
6. Broughton, T., Tan, A., and Coates, P. The use of genetic programming in exploring 3d design worlds. In *CAAD futures 1997*. Springer, 1997, 885–915.
7. Bruno, M., Henderson, K., and Kim, H. M. Multi-objective optimization in urban design. In *Proceedings of the 2011 Symposium on Simulation for Architecture and Urban Design*, Society for Computer Simulation International (2011), 102–109.
8. Caldas, L. G., and Norford, L. K. A design optimization tool based on a genetic algorithm. *Automation in construction 11*, 2 (2002), 173–184.
9. Cao, K., Batty, M., Huang, B., Liu, Y., Yu, L., and Chen, J. Spatial multi-objective land use optimization: extensions to the non-dominated sorting genetic algorithm-ii. *International Journal of Geographical Information Science 25*, 12 (2011), 1949–1969.
10. Carley, K. Sociology: computational organization theory. *Social Science Computer Review 12* (1994).
11. Celani, G., Beirão, J. N., Duarte, J. P., and Vaz, C. Optimizing the "characteristic structure": Combining shape grammars and genetic algorithms to generate urban patterns.
12. Chirkin, A. M., and König, R. *Concept of interactive machine learning in urban design problems*. ACM, 2016.
13. Coates, P., Appels, T., Simon, C., and Derix, C. Current work at ceca. In *Proceedings of the 4th Generative Art Conference (GA2001)* (2001).
14. Colton, S., Cook, M., Hepworth, R., and Pease, A. On acid drops and teardrops: Observer issues in computational creativity. In *Proceedings of the 7th AISB Symposium on Computing and Philosophy* (2014).
15. Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation 6*, 2 (2002), 182–197.
16. Derix, C. In-between architecture computation. *International journal of architectural computing 7*, 4 (2009), 565–585.
17. Doersch, C. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908* (2016).
18. Duarte, J. P., Rocha, J. M., and Soares, G. D. Unveiling the structure of the marrakech medina: a shape grammar and an interpreter for generating urban form. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 21* (2007).
19. ESRI. Esri acquires 3d software company procedural, 2011. Retrieved from: https://www.esri.com/news/releases/11-3qtr/esri-acquires-3d-software272company-procedural.html.
20. ESRI. CityEngine, 2019. http://www.esri.com/software/cityengine.
21. Farahani, R. Z., Miandoabchi, E., Szeto, W. Y., and Rashidi, H. A review of urban transportation network design problems. *European Journal of Operational Research 229*, 2 (2013), 281–302.

22. Fischer, G., and Nakakoji, K. Beyond the macho approach of artificial intelligence: empower human designers—do not replace them. *Knowledge-Based Systems 5*, 1 (1992), 15–30.

23. Flemming, U. Get with the program: common fallacies in critiques of computer-aided architectural design. *Environment and Planning B: Planning and Design 21*, 7 (1994), S106–S116.

24. Fuhrimann, L., Moosavi, V., Ohlbrock, P. O., and D'acunto, P. Data-driven design: Exploring new structural forms using machine learning and graphic statics. In *Proceedings of IASS Annual Symposia*, vol. 2018, International Association for Shell and Spatial Structures (IASS) (2018), 1–8.

25. Gero, J., and Radford, A. The place of multi-criteria optimization in design. *Design Theory and Practice, London: The Design Council* (1984), 81–85.

26. GERO, J. S. Architectural optimization-a review. *Engineering Optimization 1*, 3 (1975), 189–199.

27. Gero, J. S. Architectural optimization-a review. *Engineering Optimization 1*, 3 (1975), 189–199.

28. Gibney, E. Google ai algorithm masters ancient game of go. *Nature News 529*, 7587 (2016), 445.

29. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems* (2014), 2672–2680.

30. Groover, M., and Zimmers, E. *CAD/CAM: computer-aided design and manufacturing*. Pearson Education, 1983.

31. Hartmann, S., Weinmann, M., Wessel, R., and Klein, R. Streetgan: towards road network synthesis with generative adversarial networks.

32. Herr, C., and Kvan, T. Using cellular automata to generate high-density building form. In *Computer aided architectural design futures 2005* (2005).

33. Hillier, B. *Space is the machine: a configurational theory of architecture*. Space Syntax, 2007.

34. Horni, A., Nagel, K., and Axhausen, K. W. *The multi-agent transport simulation MATSim*. London: Ubiquity Press, 2016.

35. Janssen, P. H. An evolutionary system for design exploration.

36. Kalay, Y. E. *Architecture's new media: Principles, theories, and methods of computer-aided design*. MIT Press, 2004.

37. Koenig, R., Miao, Y., Knecht, K., Aichinger, A., and Konieva, K. Integrating urban analysis, generative design, and evolutionary optimization for solving urban design problems. *Environment and Planning B xx*, xx (2020).

38. Koenig, R., Standfest, M., and Schmitt, G. Evolutionary multi-criteria optimization for building layout planning-exemplary application based on the pssa framework.

39. Koenig, R., Treyer, L., and Schmitt, G. Graphical smalltalk with my optimization system for urban planning tasks.

40. König, R. Urban design synthesis for building layouts based on evolutionary many-criteria optimization. *International Journal of Architectural Computing 13*, 3-4 (2015), 257–269.

41. König, R., and Schmitt, G. Backcasting and a new way of command in computational design. In *CAADence in Architecture*, ETH-Zürich (2016).

42. Krawczyk, R. Experiments in architectural form generation using cellular automata. In *eCAADe* (2002).

43. Luce, R. D., and Raiffa, H. *Games and decisions: Introduction and critical survey*. Courier Corporation, 1989.

44. Machairas, V., Tsangrassoulis, A., and Axarli, K. Algorithms for optimization of building design: A review. *Renewable and sustainable energy reviews 31* (2014), 101–112.

45. Macy, M., and Willer, R. From factors to actors: computational sociology and agent-based modeling. *Annual Review of Sociology 28* (2002).

46. Malkawi, A. M. Building energy design and optimization: intelligent computer-aided thermal design.

47. Mandić, M., and Tepavŏ, B. Analysis of shape grammar application as a tool for urban design. *Environment and Planning B 42* (2015).

48. Maver, T. W. Caad's seven deadly sins. In *Sixth International Conference on Computer-Aided Architectural Design Futures* (1995), 21–22.

49. Michalek, J., Choudhary, R., and Papalambros, P. Architectural layout design optimization. *Engineering optimization 34*, 5 (2002), 461–484.

50. Michalek, J., and Papalambros, P. Interactive design optimization of architectural layouts. *Engineering optimization 34*, 5 (2002), 485–501.

51. Mitchell, W. J., Steadman, J. P., and Liggett, R. S. Synthesis and optimization of small rectangular floor plans. *Environment and Planning B: Planning and Design 3*, 1 (1976), 37–70.

52. Monizza, G. P., Bendetti, C., and Matt, D. T. Parametric and generative design techniques in mass-production environments as effective enablers of industry 4.0 approaches in the building industry. *Automation in Construction 92* (2018), 270–285.

53. Nagy, D., Villaggi, L., and Benjamin, D. Generative urban design: integrating financial and energy goals for automated neighborhood layout. In *Proceedings of the Symposium on Simulation for Architecture and Urban Design*, Society for Computer Simulation International (2018), 25.

54. Nguyen, A.-T., Reiter, S., and Rigo, P. A review on simulation-based optimization methods applied to building performance analysis. *Applied Energy 113* (2014), 1043–1058.

55. Paio, A., Reis, J., Santos, F., Lopes, P. F., Eloy, S., and Rato, V. Emerg.cities4all: towards a shape grammar based computational system tool for generating a sustainable integrated urban design. In *Proceedings of the 29th eCAADe Conference* (2011).

56. Parish, Y., and Müller, P. Procedural modeling of cities. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ACM (2001).

57. Pena, F. C., and Guerra-Gomez, J. A. Opening the black-box: Towards more interactive and interpretable machine learning. In *Proceedings of the Machine Learning from User Interaction for Visualization and Analytics Workshop at IEEE VIS* (2018).

58. Pohl, J., Myers, L., and Chapman, A. Icads; an intelligent computer-aided design environment. *ASHRAE Transactions (American Society of Heating, Refrigerating and Air-Conditioning Engineers);(United States) 96*, CONF-9006117– (1990).

59. Prusinkiewicz, P., and Lindenmayer, A. *The algorithmic beauty of plants*. Springer Science & Business Media, 2012.

60. Radford, A. D., and Gero, J. S. *Design by optimization in architecture, building, and construction*. John Wiley & Sons, Inc., 1987.

61. Schmitt, G. The impact of computer aided architectural design on physical reality. *International Journal of Architectural Computing 2*, 1 (2004), 31–41.

62. Schmitt, G., and Oechslin, W. Computer aided architectural design futures. In *CAAD Futures*, vol. 91 (1992).

63. Shi, Z., Fonseca, J. A., and Schlueter, A. A review of simulation-based urban form generation and optimization for energy-driven urban design. *Building and Environment 121* (2017), 119–129.

64. Simon, H. A. Style in design. *Spatial synthesis in computer-aided building design 9* (1975), 287–309.

65. Simon, H. A. *The sciences of the artificial*. MIT press, 2019.

66. Sindhya, K., Miettinen, K., and Deb, K. A hybrid framework for evolutionary multi-objective optimization. *IEEE Transactions on Evolutionary Computation 17*, 4 (2012), 495–511.

67. Singh, V., and Gu, N. Towards an integrated generative design framework. *Design studies 33*, 2 (2012), 185–207.

68. Spyridonidis, K.-V., and Vogiatzaki, M. Teaching and experimenting with architectural design. Tech. rep., European Association for Architectural Education, 2008.

69. Stiny, G., and Gips, J. Shape grammars and the generative specification of painting and sculpture. In *IFIP Congress (2)*, vol. 2 (1971).

70. Stouffs, R., and Rafiq, Y. Generative and evolutionary design exploration. *AI EDAM 29*, 4 (2015), 329–331.

71. Turrin, M., Von Buelow, P., and Stouffs, R. Design explorations of performance driven geometry in architectural design using parametric modeling and genetic algorithms. *Advanced Engineering Informatics 25*, 4 (2011), 656–675.

72. Von Buelow, P. Paragen: Performative exploration of generative systems. *Journal of the International Association for Shell and Spatial Structures 53*, 4 (2012), 271–284.

73. Walmsley, K., and Villaggi, L. Generative urban design, 2019. Retrieved from: https://medium.com/autodesk-university/generative-urban-design-43f4ea48039d.

74. Wetter, M., and Wright, J. A comparison of deterministic and probabilistic optimization algorithms for nonsmooth simulation-based optimization. *Building and Environment 39*, 8 (2004), 989–999.

75. Widdowson, W. *Architecture as art: a phenomenological theory of architectural esthetics*. University of Pennsylvania, 1971.

76. Wortmann, T. Model-based optimization for architectural design: Optimizing daylight and glare in grasshopper. *Technology— Architecture+ Design 1*, 2 (2017), 176–185.

77. Wortmann, T. Opossum-introducing and evaluating a model-based optimization tool for grasshopper.

78. Wortmann, T. Genetic evolution vs. function approximation: Benchmarking algorithms for architectural design optimization. *Journal of Computational Design and Engineering 6*, 3 (2019), 414–428.

79. Wright, J. A., Loosemore, H. A., and Farmani, R. Optimization of building thermal design and control by multi-criterion genetic algorithm. *Energy and buildings 34*, 9 (2002), 959–972.

80. Zitzler, E., Laumanns, M., and Thiele, L. Spea2: Improving the strength pareto evolutionary algorithm. *TIK-report 103* (2001).