VOLUME 14 No. 1, 22 December 2024 Pages 57-68

MAKE DISCOVERY THROUGH SERENDIPITY: A GENERATIVE DESIGN PLATFORM FOR PERFORMATIVE ARCHITECTURAL DESIGN EXPLORATION

MENEMUKAN KEJUTAN MELALUI KEBERUNTUNGAN: PLATFORM DESAIN GENERATIF UNTUK EKSPLORASI DESAIN ARSITEKTUR

Aswin Indraprastha
School of Architecture, Planning, and Policy Development, Institut Teknologi Bandung

Submitted: 2024-09-22; Revised: 2024-11-09; Accepted:2024-11-10

ABSTRAK

Dampak dari teknologi informasi dalam proses desain tidak dapat diragukan lagi adalah revolusioner. Adalah kenyataan bahwa generasi digital akan tidak lagi menggunakan metode konvensional atau tradisional dalam mempelajari perangkat-perangkat digital sebagai alat bantu dalam arsitektur. Hal ini menumbuhkan kebutuhan akan adanya perubahan paradigma dari teknik pedagogi tradisional ke model eksperimen (experiential learning) dimana kreativitas dan inovasi dilahirkan serta dilatih pada suatu platform yang memungkinkan adanya eksplorasi,spekulasi dan penemuan-penemuan yang tidak disengaja yang didapati dalam proses pencarian tersebut. Studi ini mencari dan menginvestigasi metode baru dalam eksplorasi desain arsitektural melalui platform desain generatif. Tujuan studi adalah sebuah platform yang memungkinkan terjadinya proses penemuan-penemuan kreatif dalam proses desain dan mendayagunakan metode generative sebagai perangkat untuk eksplorasi desain arsitektur berbasis aspek-aspek kinerjanya. Temuan studi ini mengindikasikan adanya wawasan dan metode pedagogi baru dalam pemanfaatan platform berbasis metode generative sebagai perangkat eksplorasi dan kreativitas desain arsitektur.

Keywords: *Desain Generatif; Platform Desain; Eksplorasi Desain; pedagogi dalam Desain Arsitektur; Desain Performatif.*

ABSTRACT

The repercussions of information technologies in the design process are undoubtedly revolutionary. As the digital natives will no longer require a traditional method to learn digital tools in architecture, there is a need to shift from tradition-bound techniques to an experimental mode where creativity and innovation rely upon a platform of explorative, speculative, and the recognition of serendipity and error as a credible basis on which innovation occurs. This study seeks to embrace a new method for design exploration by a generative design platform. Form-finding is encouraged through a bottom-up process of speculative actions. The goal is to cultivate serendipitous discoveries in the design process and leverage generative tools to explore performative aspects of architecture. The findings of this research offer some insights into how generative design platforms can encourage performative architectural design exploration.

Keywords: *Generative Design; Design Platform; Design Exploration; Design Pedagogy; Performative Design.*

*Corresponding author: aswinindra@itb.ac.id

INTRODUCTION

The repercussions of information technologies in the design process are undoubtedly revolutionary. The advancement of this technology disrupts the entire design process, from ideation and conceptual form exploration to facility programming, codes, and standard compliance downstream to construction documents, visualization, and communication. Kalay (2004) comprehensively introduced the accelerating transformation of architectural practice under the disruption of information technology.

In addition to those issues, the architecture profession faces a growing existential crisis regarding its relationship to automation. Automation is the only path to save the architecture profession (Deutsch, 2019). Deutsch predicts the relationship between architects and information technologies is leading to either collaborating with the technology, in which Carpo (2017) and Engelbart (2023) indicated that technology is no longer a tool for making but a tool for thinking or competing with it. Furthermore, as a tool for thinking, information technology is an integral part of the critical thinking of an architect (Deutsch, 2020).

Scholars have been investigating this new way of thinking for architectural design since the 1990s, in parallel with the development of new theories and the rise of what Schumacher argued was Parametricism 1.0 (Schumacher, 2009). The works of notable theorists and practitioners such as Frank Gehry, Zaha Hadid, and Kas Oosterhuis were examples of the realization of this new thinking and tools.

Specifically, one of the characteristics of the design process is the bidirectional interaction between the designer and the object in the exploration of design solutions, which has been an intensive topic for research. For example, Killian (Axel Killian, 2006) developed an interactive design exploration model to bridge ideas and their representation. His study used the computational model as a sandbox - an exploration and interaction platform.

Davis (Daniel Davis, 2013) investigated the relationships between parametric software and the flexibility of parametric modeling. Flexibility in the context of Davis's study is the capability of a parametric model to support multi-modal thinking and action in the design processes that produce variations, cyclic iterations, and trial and error. Those exemplary studies began with the awareness that software is an inseparable part of design activity and significantly influences design thinking.

Vice versa, architectural design thinking and design actions should include this new paradigm, knowledge structure based on computing (Hensel et al., 2013), (Oxman, 2006), (Woodbury, 2010) as well as predicted by Schön (1983) and Negroponte (1973) in the early 70s.

Furthermore, in the academic context, nurturing and fostering creative thinking and action has become the keystone of architecture education. Creative thinking is the cognitive ability to develop unique, original, and meaningful design solutions by consistently training the brain to explore new connections with new information. In his book, Johnson (Steven Johnson, 2010) referred to the experiments conducted by brain scientist Robert Thatcher (2007), where creativity in the form of new connections in the brain cells can only be achieved if neuron signals are used to search and explore new and unfamiliar information. The experiment concluded that discovering something meaningful, unpredictable, and unforeseeable output through search and exploration is a creative journey that can be trained. This finding gives insight into the potential utilization of the computational model as a platform for creative exploration and fostering discovery through a series of trial-and-error activities.

This study unravels a new approach to support creative exploration through the fundamental concept of generative methods in design processes, which has rarely been studied, especially in academics. This study offers a new direction in parametric design pedagogy to open the potential to create a platform that fosters serendipitous connections and helps nurture creativity.

Literature review

Learn to a Meaningful Discovery

In today's generation, where information and knowledge seem ubiquitous with the proliferation of online media and peer-to-peer learning from others, learning is a more effective way to develop cognitive, motoric, and affective abilities than teaching. The teaching method in the traditional sense of the authoritative role of the teacher, one-directional pedagogical style, and passive mode of inquiry are no longer suitable for the digital natives. Instead of drawing and modeling, computation has the potential for exploration, experimentation, or making speculative propositions, as Kolarevic proposed, shifting from the making of form to the finding of form (Kolarevic, 2004). Students are exposed to the platform and environment where they learn that innovative ideas occur in an incremental, cumulative, and sometimes not intended creative process. According to the findings of experiments by Johnson and Vermillion, digital natives learn by embracing accidents and making discoveries through serendipity (Johnson & Vermillion, 2016).

Generative Design Model

Generative models of computational design are characterized by providing computational mechanisms for formalized generation processes (Oxman, 2017, 2006). According to Oxman, bi-directional interactions between the designer and the design object play a significant role in this model (see also Killian's and Davis's thesis). In generative design models, the complex system resulting from generative mechanisms is often arbitrary, unpredictable, and more than the sum of its parts. This contrasts with the authoring model, where the computation mechanism is used as a geometrical and topological authoring tool, i.e., drawing and modeling. As proposed by Oxman, the generative design model provides an interactive and creative interaction of form-finding between the designer and the digital representative of the design output, as illustrated in Figure 1 below.

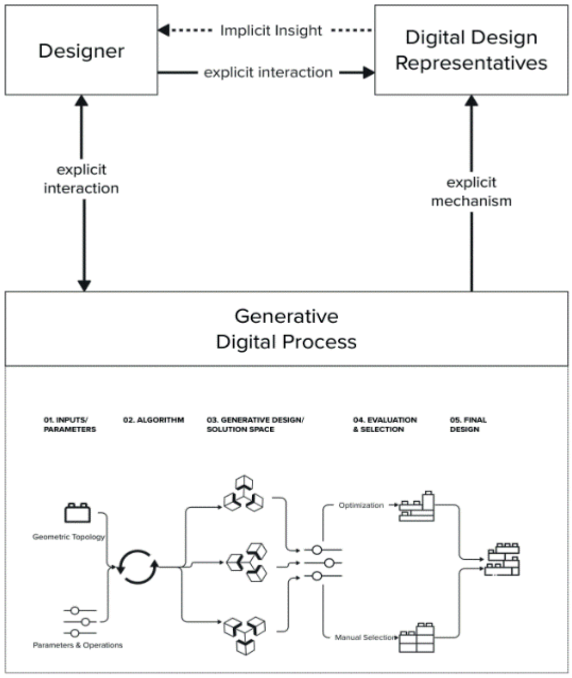


Figure 1.
Generative Model Scheme
Source: Modified from Oxman (2006)

Previous Study

Despite the potential to be a novel strategy that allows architects/designers to take advantage of computational processes and nurture innovation and creativity by exploring design solutions, the generative design field in the architectural design process, particularly in education, is still under development and partially explored. Scholars experimented with frameworks and workflows utilizing generative methods in a design process. For example, Terzidis (2006) experimented with what he advised as an inductive algorithm, which is not tailored to automate tasks or aimed at predictable results. This inductive algorithm strategy addresses and explores generative processes and simulates complex phenomena. Another different approach was a study conducted by Bollmann & Bonfiglio (2013), where generated designs were produced by a framework of Design Constraint Systems (DCS) utilizing combinatorial mechanisms. This system has a two-component approach to generating designs; the first component describes the abstract structure of the modeled objects, while the second component interprets the structure and generates the actual geometric forms.

Johnson J. & Vermillion J. (2016) introduce a generative algorithm in its basic definition by generating geometries through a series of bottom-up processes and using the basic building blocks of points, lines, and simple operational sequences to develop a series of assemblages or complex systems.

Moreover, Mukkavaara & Sandberg (2020) proposed a framework of architectural design exploration using generative design that highlighted open, generative algorithms and elaborated exploration mechanisms for the solution space in a top-down fashion. This framework was demonstrated for the early conceptualization stage with a residential block as a case study. Another study that attempted to utilize generative methods in the design process was by Ashari et al. (2022), with the preposition that the generative method has the automatic mechanism to produce design iterations in a shorter time than conventional 3D modeling. Rather than generating variants of design candidates, the study focused on using preset scripts of simplified generative algorithms to generate a complex design and analysis outcome. This study takes a different approach based on previous attempts to utilize the generative method in the design process. It focuses on building skills in developing and adapting procedures for generating patterns, shapes, and forms.

METHOD

Our study on generative platform for design exploration is based on an integrated generative design framework proposed by Gu et al. (2010) with a focus on explorative and building skills in developing complex systems out of simple shapes and operations as previously studied by Johnson & Vermillion (2016). With exploration and discovery through accident, trial, and error, or serendipity in mind, we experimented through a series of exercises in a parametric design platform.

The study design follows the procedure as follows:

1. Develop a research framework in which the objective of the study can be achieved through the lens of three case studies with incremental complexities of input, logic and processes, and algorithms.
2. Develop the experiment procedure and mechanism of each case study using a bottom-up approach, which means it will start with an initial unit of geometry, observable transformative algorithms, and an iterative/loopback mechanism.
3. Determine courses for the experiment where each case study will be implemented as student assignments. Each case study defined its achievable goal for the final output (see Figure 2).
4. Determine the subject of the experiment for each case study, which is the student of the courses.

Figure 2 shows a computational framework to generate a complex system from basic shapes used for three case studies. This framework uses a bottom-up approach where students start with limited shapes and constraints. In experiments with computational design thinking, a parametric design platform was used using Rhinoceros 3D and Grasshopper as Visual Algorithm Editor (VSA). The basic workflow, functions, and data structure in VSA were incrementally introduced as part of the course, which aims to deliver a holistic comprehension of parametric design and diminish the production with-out comprehension workflow, as argued by Senske (2014).

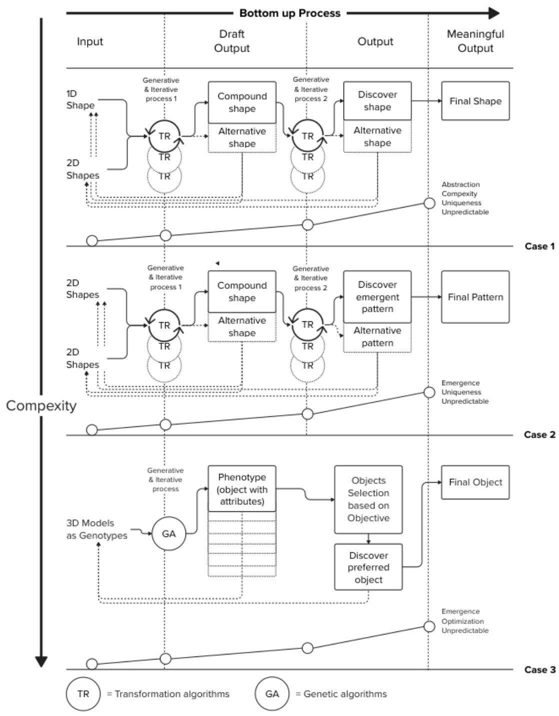


Figure 2.
A framework of generative method for design exploration
Source: Author Analysis (2024)

The participants were 2nd-year undergraduate students of Architecture with no prior computer programming background other than the mandatory course of Introduction of Computing in their first year. The total number of the participants is 95 students. The course comprises lectures, practice, and tutoring sessions, and three academic credits are equivalent to nine hours of academic activities per week.

Instead of going for the authoring direction where the parametric method is used to produce a model out of abstraction, we trained students in the principles of object generation and the underlying mechanism for producing multiple objects in a solution /latent space. We introduced the shift from the concept of form and its representation to the concept of formation, i.e., the performative process and mechanism of form generation and self-organization as propositioned by Hensel and Menges (Hensel et al., 2013, 2004) and Oxman (2006) (in Peteinarelis and Yiannoudes (2018). As proposed, three case studies were conducted to represent incremental strategies of generative methods for design exploration and discovering solutions out of serendipity during its processes.

Case Study 1: Algorithmic Generation

The objective of the first case study is to demonstrate an understanding of the basic algorithmic and generative processes utilizing sequential transformations on a shape. The design brief is to generate an algorithmic shape resulting from the generative mechanism with given constraints. Students were required to expand a prescribed simple algorithm and attempt to create a complex shape that was difficult to discern from its parts.

The basic shapes consisted of three types of shapes:

1. 3-4 segments of Open NURBS spline curve,
2. 3-4 segments of the Open NURBS curve, and
3. 3-4 segments of a Closed NURBS curve.

While the transformative operations are move/copy, rotate, scale, and mirror. The final generated shape should consist of a total of 40 compound shapes.

This exercise aims to generate a complex shape/system represented by a generative pattern where the designer (student) has control of the input and rules while the algorithm determines the output. While expanding the algorithm and experimenting with the sequential transformation mechanism, students can create and control some parameters and objects' positions relative to other objects. The final output will be chosen and discovered intuitively based on the aesthetic value of the complex shape. As indicated in Figure 3, different parameters and sequential rules can generate different final outputs to suit the student's preferences.

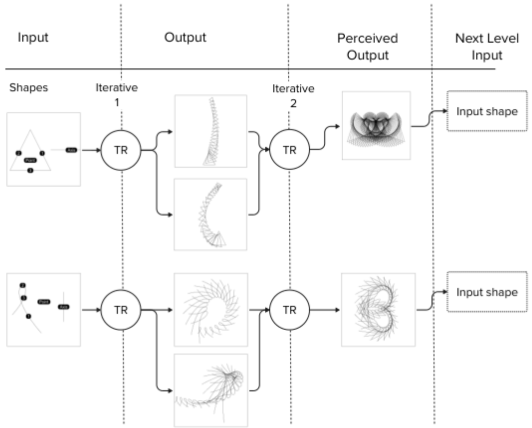


Figure 3.
An Example Scenario of Sequential Transformations in Case Study 1
Source: Author Analysis (2024)

Case Study 2: Emergent Pattern

In the second case, students can reuse and expand their basic algorithms to generate emergent patterns on a grid. Instead of generating a single and complex shape, students examined and investigated emergent patterns generated by a series of transformations of shapes over a grid. The transformed shapes over each row or column on a grid will generate a compound configuration or unimagined field beforehand. This field is the goal of this case, where students experimented with 2- dimensional transformation rules to find patterns that appear on the grid with visual effects. Students experimented using three basic shapes:

1. 3-4 segments of Open/Closed NURBS curves,
2. four 3-4 segments of Open NURBS curves, and
3. four 3-4 segments of Closed NURBS curves,

wherein each type of shape student can reuse and expand the previous algorithm. Adding complexity in this case, the generated pattern on the grid shall produce asymmetry or less figurative shape as might be produced in the previous case. The algorithm that re-executes over shapes on a grid is the field generator, as depicted in Figure 4.

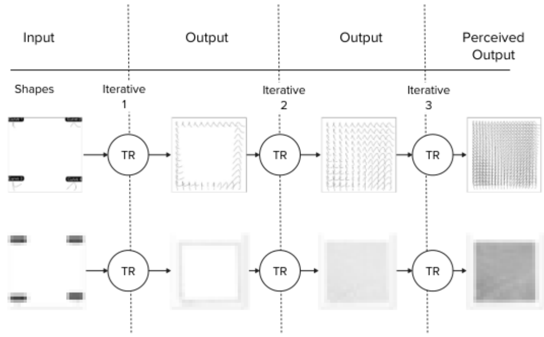


Figure 4.
An Example Scenario of Sequential Procedures on a Grid in Case Study 2
Source: Author (2024)

Case Study 3: Explorative Design

In the third case, students were requested to employ generative algorithms to produce a series of design alternatives or solution candidates that comply with simple objective criteria. This deterministic approach was introduced as a technique for using generative methods for design exploration with control or evaluation mechanisms.

This case study aims to enhance the fundamental concept of the generative method using the genetic algorithm for design exploration, as well as the principal understanding of the generative, evaluative, and visualization processes. Students started with generative design problems, whether design conceptualization, a conceptual massing design, or design exploration of architectural elements in conjunction with their performances. Generative algorithms employ genetics algorithms or combinatorial transformation rules based on parameters.

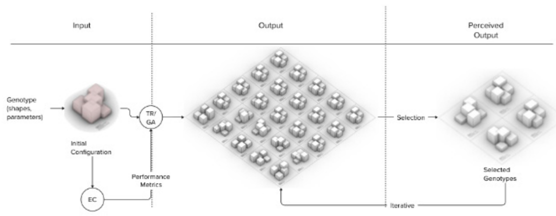


Figure 5.
An Example Scenario of Generative and Evaluation Criteria Mechanism on a Grid in Case Study 3
Source: Author (2024)

Figure 5 depicts a scenario where generative and evaluation criteria were used to select and filter solution candidates of massing configurations along with the metrics attributes. This process allows the exploration of solution/latent space and the intuitive discovery of preferred design solutions by the spatial configuration and their performance metrics.

RESULTS AND DISCUSSION

Discovery Mechanisms as Creative Operation

Scholars have studied and made models of creative processes. Guilford's Structure of Intellect Theory (Carroll & Guilford, 1968), Hayes' five cognitive processes in creativity (Hayes, 1989), and Stenberg's Propulsion model (Sternberg, 2003) are examples of some creative models. In the context of this study of identifying problems and exploring solution spaces in the creative process, we examined the scheme by Stempfle and Badke-Schaub (2002), which had defined four basic cognitive operations: *generation* and *exploration* for divergence and *comparison* and *selection* for convergence.

According to the scheme, divergent thinking is where generation and exploration occur and widen the scope of possible solutions. In the parametric–generative design method, the scripted algorithm, parameters, and operations define the level of divergence and opening process for discovery through exploration. Moreover, as Lee & Ostwald, (2020) suggested, the parametric design method supports two creative processes: divergence by parameters and convergence by rules. In this scheme, we used categories of creative operations adapted from Sternberg (2003). The activities of redefinition and forward incrementation are the divergent stages of creative operation, and redirection and reconstruction are the convergence stages, as depicted in Table 1.

Evaluation Criteria

Given the research design and objectives, the evaluation criteria are developed based on divergent and convergent stages of thinking and action. The discovery moment at the convergence stage is a cognitive-intuitive action made possible by the interplay between parameters, operations, and output objects (see Oxman's Generative Model).

Table 1.
Creative operations and generative method description

Creative Operation	Generative Method Description
Divergence (P-Process): Redefinition & Forward Incrementation	explicit abstraction & problem decomposition (A), parameters determination (R), rules & transformations (T)
Convergence (O-Output): Redirection & reconstruction	Surprisingness (S), Emergence (E), Degree of variability (V), Complexity (C)

Source: Author Analysis (2024)

Table 1 describes a matrix of two creative operations with the role of generative methods. In the divergence stage, ideas and imagined goals are constructed through the parametric process of redefining algorithms and iterative processes of interconnecting relations between operands and operators. Rather than constructing a goal or objective, this process aims at connecting possibilities and incremental transformation to see the "what if ."The generative method in this stage is utilized to formalize given problems or constraints and produce possibilities rather than a precise output using a set of rules and parameters.

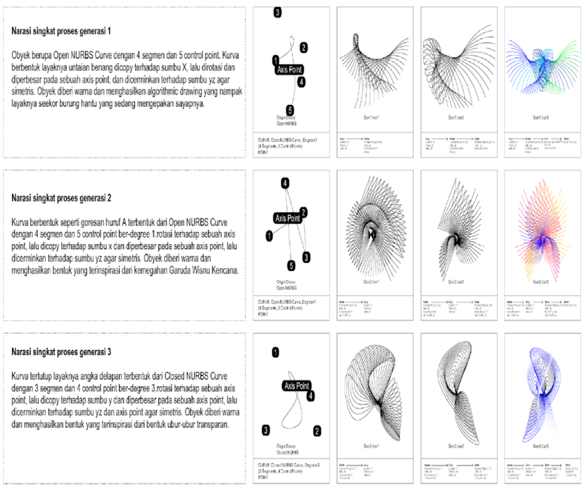
Consequently, the convergence stage is an optimization or selection process based on intuitive and subjective judgment for the given algorithm output. After evaluating previous studies, we defined four characteristics of the output: Surprisingness, EmerEmergentgree of Variability, and Complexity.

Analysis of Experiments

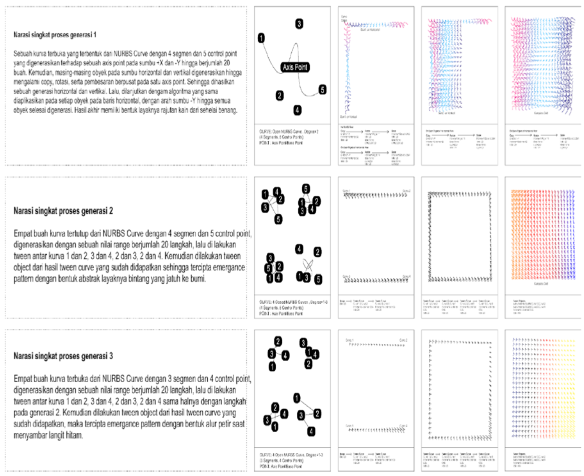
The results sample from 20 students in each of the three case studies were analyzed using two approaches:

1. The divergent stage is analyzed based on objective aspects of generative components and subjective aspects of redefinition/abstraction and problem decomposition. Objective generative components are the number of parameters (R) and transformative operations (T) that were used in the algorithms, whereas subjective redefinition is an explicit elaboration of problem statements and steps to be undertaken to solve the problem and to be written in the algorithms (A).
2. The convergence stage is analyzed based on subjective metrics. Subjective metrics of the final outputs are Surprisingness (S), Emergence (E), Variances (V), and Complexity (C). All subjective metrics of each output are evaluated by different lecturers using 1-5 scales where 1 is the minimum and 5 is the maximum.

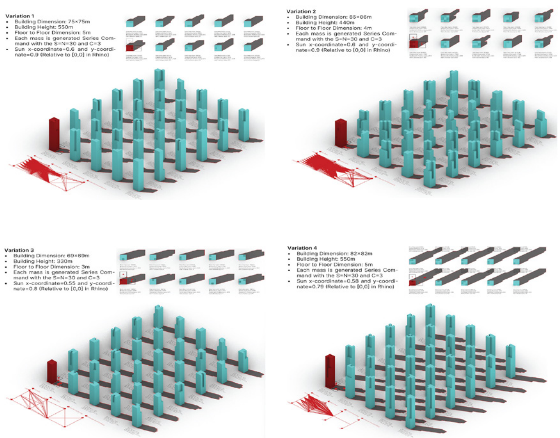
Figure 6 illustrates samples of three cases of generative exercises by the students.



Case 1



Case 2



Case 3

Figure 6.
Sample of Processes and Generated Objects by Students in Case Studies 1, 2, and 3
Source: Author (2024)

The performance metrics of the output Y (convergent stage) then is the function of X (divergent stage) as formulated below:

$Y_{(\text{dependent var.})} = O_{[S, E, V, C]}$; Output as performance function of S, E, V, C attributes.
$X_{(\text{independent var.})} = P_{[A, R, T, O]}$; Process as a function of A, R, T attributes.

We examined the relationship between Y and X to analyze significant factors of generative methods for creating space for discovery, as summarized in Table 2 below.

Table 2.

Case 1

	Process				Output		
	P_A	P_R	P_T	Mean $P_{(A,R,T)}$	O_S	O_C	Mean $O_{(S,C)}$
n=20							
Max	1,00	1,00	1,00	0,89	1,00	1,00	1,00
Min	0,60	0,27	0,55	0,52	0,75	0,63	0,69
Mean	0,73	0,59	0,68	0,67	0,87	0,86	0,87
Stdev	0,14	0,19	0,11	0,10	0,12	0,12	0,10

where:
P_A : *Declarative Abstraction (1= less elaborative; 5=elaborative);*
P_R : *Avg. num. Total Parameters;*
P_T : *Avg.num. Total Transformations Steps.*
O_S : *Surprisingness (1=less surprising; 5=surprising);*
O_C : *Complexity (1=less complex; 5=complex).*

Case 2

	Process				Output			
	P_A	P_R	P_T	Mean $P_{(A,R,T)}$	O_E	O_V	O_C	Mean $O_{(E,V,C)}$
n=20								
Max	1,00	1,00	1,00	0,92	1,00	1,00	1,00	1,00
Min	0,40	0,11	0,61	0,42	0,63	0,71	0,69	0,67
Mean	0,69	0,54	0,80	0,68	0,85	0,85	0,84	0,84
Stdev	0,16	0,23	0,11	0,11	0,12	0,10	0,09	0,09

where:
P_A : *Declarative Abstraction (1= less elaborative; 5=elaborative);*
P_R : *Avg. num. Total Parameters;*
P_T : *Avg.num. Total Transformations Steps.*
O_E : *Emergent Pattern (1=minimal; 5=maximal);*
O_V : *Pattern Gradient (1=less obvious; 5=obvious);*
O_C : *Complexity (1=less complex; 5=complex).*

Case 3

	Process				Output		
	P_A	P_R	P_T	Mean $P_{(A,R,T)}$	O_V	O_C	Mean $O_{(V,C)}$
n=20							
Max	1,00	1,00	1,00	1,00	1,00	1,00	1,00
Min	0,60	0,57	0,57	0,71	0,60	0,60	0,60
Mean	0,84	0,81	0,81	0,82	0,73	0,74	0,74
Stdev	0,16	0,15	0,16	0,08	0,15	0,14	0,12

where:
P_A : *Declarative Abstraction (1= less elaborative; 5=elaborative);*
P_R : *Avg. num. Total Genotypes;*
P_T : *Avg.num. Total Metrics.*
O_V : *Variability (1=low diversity; 5=high diversity);*
O_C : *Complexity (1=less complex; 5=complex)*

Summary Results

In three case studies, divergent operations were determined by factors such as:

1. [Subjective] Declarative / explicit abstraction: explicitly explaining the problems and describing the steps to achieve the goal sequentially.
2. [Objective] Set up parameters/values that generate variability (variance) of the outputs.
3. [Objective] Set up operation mechanisms and metrics to control the outputs.

Whereas the convergent operations through the generative method were determined by:

1. [Subjective] Surprisingness: the degree of unpredictable outputs;
2. [Subjective] Emergence: the viability of emergence patterns;
3. [Subjective] Variability: the degree of variability of the outputs;
4. [Subjective] Complexity: the degree of complexity of the outputs.

All subjective metrics are measured through a Likert scale of 1 to 5, where one is the minimum, and five is the maximum.

The divergent operation through A, R, and T plays a significant role in producing a solution/latent space that yields the discovery of the final output. Figure 7 shows the P-Process (A, R, T) data distribution of case studies. This indicates that the problem decomposition and algorithm used to achieve the goal were elaborated with a median of 0.6 (case 1) and 0.8 (case 3), respectively. In particular, case 2 - *Emergent Pattern* has the most distributed data (Standard Deviation=0.11), which means that various techniques (R, T) in the algorithm are used to generate the emergence pattern.

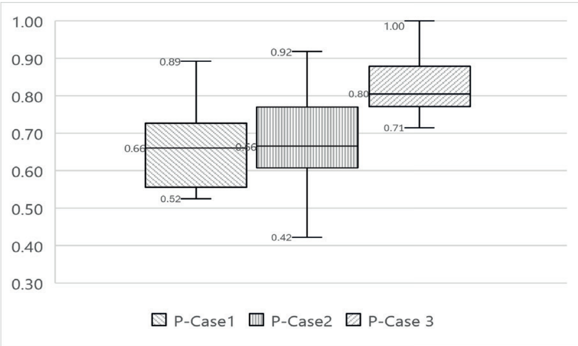


Figure 7.
Distribution Plot of P-Process of Case Studies 1, 2, and 3
Source: Author (2024)

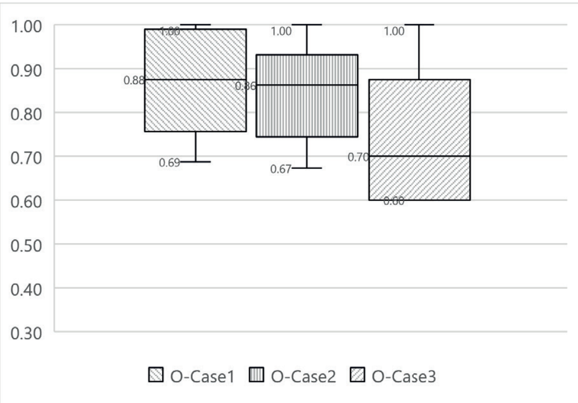


Figure 8.
Distribution Plot of O-Output of Case Studies 1,2, and 3
Source: Author (2024)

Moreover, Figure 8 of O-Output shows a wider IQR than the P-Process, indicating more dispersed subjective evaluation on the attributes of the output (S, E, V, C) with standard deviations of 0.1, 0.09, and 0.12 on case 1, case 2, and case 3 respectively. This means that generated outputs in the solution space yield many variabilities. Particularly in case 3, which employed the combinatorics principle for the generative method, it generates a vast solution space.

Table 3.
Correlation Results

Case 1	O_S	O_C	
P_A	**0.56**	0.39	
P_R	0.50	0.08	
P_T	0.05	-0.18	
Case 2	O_E	O_V	O_C
P_A	**0.41**	-0.15	**0.10**
P_R	0.06	**0.27**	-0.21
P_T	-0.10	0.01	0.00
Case 3	O_V	O_C	
P_A	0.12	0.10	
P_R	**0.32**	0.21	
P_T	0.18	0.08	

Source: Author Analysis (2024)

Table 3 shows the contribution of each of the Process attributes (A, R, T) to the Output metrics (S, C, E, V), indicating that Declarative/explicit abstraction (A) as a starting point for the generative method has significant impact shaping solution space. Furthermore, the parameters (R) are vital contributors to the generation of diverse yet meaningful solution candidates in solution/latent space.

CONCLUSION

The study explored a new method for performative architectural design exploration by a generative design platform. The experiment of three case studies proves that the generative platform has the potential for a creative operation that leads to discovery moments during the design process. By utilizing generative design platforms, architectural students can engage in a creative and exploratory design process that has space for trial and error and allows for innovative and creative solutions.

The finding shows the importance of explicitly declaring the problem statement and problem decomposition as significant factors in preparing generative algorithms and setting up solution/latent space.

The generative method's role as a sandbox platform is crucial to allowing explorative action, nurturing divergent thinking and a trial-and-error mindset, and providing insight into expected or unexpected output that can be selected deliberately for further iterative design.

This finding confirms the principle of computational thinking, which involves problem decomposition and abstraction before developing solutions through algorithms.

Further study on large samples with various degrees of computational skills is required to extend comprehension, and it is crucial to gain a deeper understanding of the generative method's effectiveness as a platform for architectural design exploration.

This expanded study could provide valuable insights into how different computational skill levels and backgrounds impact the adoption and utilization of generative design tools in the architectural design process.

BIBLIOGRAPHY

Ashari, A., Amoroso, N., Kelly, S. (2022). Generative Design in Landscape Architecture: Defining Three Design Scripts for Beginners. *Journal of Digital Landscape Architecture*. https://doi.org/10.14627/537724058

Axel Killian. (2006). Design exploration through bidirectional modeling of constraints. *Thesis*. Massachusetts Institute of Technology, Dept. of Architecture, MIT, Massachusetts.

Bollmann, D., Bonfiglio, A. (2013). Design constraint systems - A generative approach to architecture. *International Journal of Architectural Computing 11*. https://doi.org/10.1260/1478-0771.11.1.37

Carpo, M. (2017). The Second Digital Turn: Design Beyond Intelligence, Writing architecture series. *Paperback, The MIT Press*. ISBN: 9780262534024

Carroll, J.B., Guilford, J.P. (1968). The Nature of Human Intelligence. *American Educational Research Journal*. https://doi.org/10.2307/1161820

Daniel, D. (2013). Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture. *Thesis* for Doctor of Philosophy. School of Architecture and Design, RMIT, Melbourne.

Deutsch, R. (2020). Design Thinking in Architecture, in: Think Like an Architect. *Routledge.* https://doi.org/10.4324/9781003108535-43

Deutsch, R. (2019). Superusers: Design technology specialists and the future of practice, Superusers: Design Technology Specialists and the Future of Practice. *Routledge.* https://doi.org/10.4324/9781351138987

Engelbart, D.C. (2023). AUGMENTING HUMAN INTELLECT: A Conceptual Framework, in: Augmented Education in the Global Age: Artificial Intelligence and the Future of Learning and Work. *Routledge*. https://doi.org/10.4324/9781003230762-3

Gu, N., Singh, V., Merrick, K. (2010). A framework to integrate generative design techniques for enhancing design automation, in: New Frontiers - *Proceedings of the 15th International Conference on Computer-Aided Architectural Design in Asia*, CAADRIA 2010. https://doi.org/10.52842/conf.caadria.2010.127

Hayes, J.R. (1989). *Handbook of Creativity. Perspectives on Individual Differences*. Plenum Press, New York.

Hensel, M., Menges, A., Weinstock, M. (2013). Morphogenesis and Emergence (2004□2006), in: *The Digital Turn in Architecture* 1992–2012. https://doi.org/10.1002/9781118795811.ch11

Hensel, M., Menges, A., Weinstock, M. (2004). Emergence in Architecture. *Architectural Design* Vol. 74, 3rd Ed.

Johnson, J.S., Vermillion, J. (2016). Digital design exercises for architecture students, Digital Design Exercises for Architecture Students. *Routledge.* https://doi.org/10.4324/9781315742229

Kolarevic, B. (2004). Architecture in the digital age: Design and manufacturing, Architecture in the Digital Age: Design and Manufacturing. *Taylor & Francis.* https://doi.org/10.4324/9780203634561

Lee, J.H., Ostwald, M.J. (2020). Creative decision-making processes in parametric design. *Buildings Journal.* Vol.10, 12th Ed. https://doi.org/10.3390/buildings10120242

Mukkavaara, J., Sandberg, M. (2020). Architectural design exploration using generative design: Framework development and case study of a residential block. *Buildings Journal.* Vol. 10, 11th Ed. https://doi.org/10.3390/buildings10110201

Negroponte, N. (1973). The Architecture Machine: Toward a More Human Environment, Computer-Aided Design. *MIT Press.* https://doi.org/10.7551/mitpress/8269.001.0001

Oxman, R. (2017). Thinking difference: Theories and models of parametric design thinking. *Design Studies*. Elsevier Vol. 52. https://doi.org/10.1016/j.destud.2017.06.001

Oxman, R. (2006). Theory and design in the first digital age. *Design Studies.* Elsevier Vol. 27. https://doi.org/10.1016/j.destud.2005.11.002

Peteinarelis, A., Yiannoudes, S. (2018). Parametric Models and Algorithmic Thinking in Architectural Education, in: *Proceedings of the International Conference on Education and Research in Computer Aided Architectural Design in Europe.* https://doi.org/10.52842/conf.ecaade.2018.2.401

Schon, D.A. (1983). The Reflective Practitioner: How Professionals Think in Action (Arena). *Basic Books*. https://doi.org/10.4324/9781315237473

Schumacher, P. (2009). Parametricism: A new global style for architecture and urban design. *Architectural Design*. https://doi.org/10.1002/ad.912

Senske, N. (2014). Confronting the challenges of computational design instruction, in: Rethinking Comprehensive Design: Speculative Counterculture - *Proceedings of the 19th International Conference on Computer-Aided Architectural Design Research in Asia*, CAADRIA 2014. https://doi.org/10.52842/conf.caadria.2014.821

Stempfle, J. Badke-Schaub, P. (2002). Thinking in design teams - An analysis of team communication. *Design Studies*. Vol. 23. https://doi.org/10.1016/S0142-694X(02)00004-2

Sternberg, R.J. (2003). Creativity and Development; *Oxford University Press*, New York.

Steven, J. 2010. *Where Good Ideas Come From, the Natural History of Innovation*, 2010th ed. Riverhead Books, New York.

Terzidis, K. (2006). *Algorithmic architecture, Algorithmic Architecture*. Books: Routledge. https://doi.org/10.4324/9780080461298

Woodbury, R. (2010). *Elements of parametric design. Book*: Routledge. ISBN 9780415779876

Yehuda E. Kalay. (2004). Architecture's New Media: Principles, Theories, and Methods of Computer-Aided Design. *Nexus Network Journal*. MIT Press. DOI:10.1007/s00004-005-0015-1