Cellular Automata as a Teacher of Machine Learning: A Human-machine Collaboration to Contextualise Environmental Architecture

Provides Ng [1,2], David Doria [2], Nikoletta Karastathi [2], Carlos Rivera Salaverry and Alberto Fernandez [2]

Abstract. The challenge to creating buildings that intelligently respond to local environmental conditions while respecting cultural needs is increasingly dependent on hybrid computational workflows that are flexible and adaptive to designers' creative decisions and needs. We propose a novel approach where Cellular Automata (CA)—simple digital building blocks that evolve via programmed rules—act as teachers to Machine Learning (ML) systems, which learn patterns from data to automate and accelerate tasks. Our framework uses CA states to translate site-specific intelligence (e.g., solar instances or cultural landmarks) into training protocols for ML, enabling rapid generation of context-sensitive solar designs. Through workshops with 20 architects across 12 countries, we tested the framework and results showed how CA successfully encoded environmental constraints into ML and significantly reduced computation time. However, the pipeline required significant human refinement and spontaneity to resolve computational bottlenecks, highlighting human-machine collaboration. Critically, we argue that automation's value lies not in replacing designers but in structuring a pedagogical collaboration: Humans define contextual priorities, CA formalises them into teachable rules, ML accelerates application. Our work ultimately questions whether intelligence can scale without collaboration, offering technology not as a solution but as a framework for negotiating this tension.

Keywords: artificial intelligence; bio-inspired design; solar architecture; discrete computation; human-machine interaction

[1] The Chinese University of Hong Kong
[2] University College London (UCL)
e-mail: provides.ism@gmail.com

1 Introduction

Machine Learning (ML) algorithms are the applications through which Artificial Intelligence (AI) is achieved. In other words, if AI describes an intelligent computer program, ML comprehends the algorithms that allow such intelligence to exist. These algorithms usually learn by progressively improving as it iterates over a large dataset, or contexts that provide this data. Through the iterative process, ML algorithms tune itself to achieve the expected behaviour, such as finding deep patterns across data points, or predicting future states of data, or even predicting how it should behave in an unseen scenario. This is what allows it to automate intellectual tasks.

As technology quickly progresses, it is not easy to determine what ML algorithms are and will be able to achieve, but one can say that the current application of these is overall limited not simply by hardware or cloud capacity, but creative implementation methods that can overcome such limitations; that said, the development and application of ML algorithms requires cautious reflections. Algorithms will carry over the biases that are present in the data it consumes, or the crafted environments it learns from. Consequently, the process of selecting or creating the data that a ML system will consume is a crucial step in designing its application. A dataset that is not carefully constructed can lead to a training process that outputs an unfit model, or, in social applications, reinforce prejudices and problematic behaviours. This research, however, seeks to use biases as an advantage – it seeks to embed specific bias in the creation of custom datasets as a means of formulating human-machine interaction.

The project's premise is to focus on the creation of the dataset, with the help of other synthetics intelligences, such as Cellular Automata (CA) to determine the ML algorithms behaviour. CA is a discrete computational model that is well-suited to simulating contextual information to a certain extent because they explicitly model local interactions between cells based on their neighborhood states. It is often used in architecture design to simulate complex patterns through controlled input, such as solar radiation. By embedding the characteristics of such rule-based, computationally demanding processes into the dataset that will be used for training, the objective is to capture into a ML model a design process that, originally would take several hours, can be executed relatively instantaneously.

The expected result is an algorithm that is capable of predicting 3-dimensional shapes in a given urban context, that could be used to populate architectural elements in a manner that maximises solar incidence, with the end goal of also maximising the capability of a structure to harvest energy from photovoltaic elements or through passive solar strategies. ***How can rule-based systems translate contextual information to teach AI in automating repetitive and computationally heavy tasks, and synergistically enhance human creativity in sustainable architecture?***

2 Literature Review

Generative Adversarial Networks (GANs) is a class of ML models designed for generative tasks, which was quickly adopted by architectural design disciplines, as it presents a novel approach to automate image generation, from building layouts to spatial renderings (Chaillou, 2019). However, as GANs operate on pixel data, questions arise over how it can be embedded with relevant contextual information beyond simple form-finding, such as environmental and cultural considerations, to ensure that the generated designs are contributing to sustainable development goals. Recent advances in bio-inspired computational design highlights the critical tension of how ML algorithms can synergistically enhance human creativity, with three key debates.

First, The Digital Universalism in Computational Design. While Frazer's (2002) evolutionary architecture prioritized top-down optimization, contemporary research leverages bottom-up biological intelligence for sustainability. For instance, Ertan and Adem (2024) showed how discrete aggregation of timber units can encode craftsmanship, presenting data aggregation as a form of cultural expression, responding to Charitonidou's (2022) critique of "digital universalism".

Second, Translating Theoretical Automata to Design Agents. While classical Cellular Automata (CA) theory (Wolfram, 2005) emphasized emergent complexity from simplicity, recent work integrates environmental data. Reimagined discrete aggregation discourses (Retsin, 2019; Kohler, 2017) prioritised ecological and topological assembly, but discounted evaluating the knowledge transfer between algorithms.

Third, Data Biases as Design Opportunities. ML's rise intensified debates about agency and contextual sensitivity. Work exploring the relationship between text, images, and form in ML (Koh, 2020; Bolojan, 2021) established linguistic and visual principles for stylistic exploration; simultaneously, it highlighted the role of unintentional factors or biases in the learning process. This reframed operational bias as a vehicle for embedding priorities into a dataset.

Across these cases, discrete computation was the trigger for complexity via simplicity. In the case of CA, the outcomes are based on the interplay of basic growth rules in a strict relationship with an environment. The best feasible solution to the interaction of these components in an n-dimensional space is the outcome. As a result, such aggregation logic favours choices above ideal solutions in a dynamic process that is always changing—Dynamism. Therefore, working with aggregated shapes and adding characteristics that define a CA, we may produce a spectrum of outputs that speculatively inform the solution space, with seeds acting as agents and geometrical outputs acting as aggregation rules. This allows for the testing of different data inputs, such as solar radiation to enhance a building's light-harvesting quality and iterate options in an evolutionary manner.

This presents an opportunity to synthesize cross-intelligence systems for tailored cases, responding to specific disciplinary needs. This study uses CA to train ML models for architectural acceleration and contextualisation—a gap that our "CA-as-teacher" pipeline addresses. This positioned CA not as a form-generator but as a bridge between designer intuition (rule definition) and ML efficiency (pattern replication), shifting CA from a computationally-heavy application to an intelligent pedagogical tool for ML.

3 Methods

Addresses the research gap between discrete computation, biomimicry, and ML contextualization in architectural design, this research responds to limitations identified in our prior CA-to-GAN pipeline (Ng et al., 2021; 2022), which encountered three critical challenges:

1. Rigid discretization hindered organic form and cultural adaptation
2. Inadequate control for contextual variables (local resources, aesthetics)
3. Computational bottlenecks in translating CA outputs to design solutions

To resolve these, we developed a novel three-phase methodology.

PHASE 1: CA-ML Pipeline Enhancement with Houdini. Houdini's procedural environment enabled three key innovations, including biologically-grounded growth simulation, adaptive discretisation with dynamic voxel resizing (5-50cm^3) responding to solar incidence thresholds, and data standardization with lossless translation between CA states to generate adaptive design outputs.

PHASE 2: Cross-context Deployment. A series of workshops were organised where 20 designers from 12 countries worked in teams of 4 to test the pipeline, following a task protocol.

1. Select hometown context with documented climatic/cultural constraints
2. Generate climatic data using CA to train GANs
3. Translate CA states into Houdini to create baseline design solutions
4. Iterate designs with human-GAN feedback to accelerate the process

PHASE 3: Benchmarking. Comparing rule-based and ML workflows in terms of time per generation, total computational time, and data volume.

Table 1 The algorithmic components and specification of the proposed hybrid workflow.

Component	Specification	Role
Houdini 19.5	VEX-based growth algorithms	Bio-inspired form generation
Rhino 3D	Galapagos evolutionary solver	Solar optimization
PyTorch	Custom pix2pix	ML acceleration
RTX 4090	48GB VRAM	Training/inference

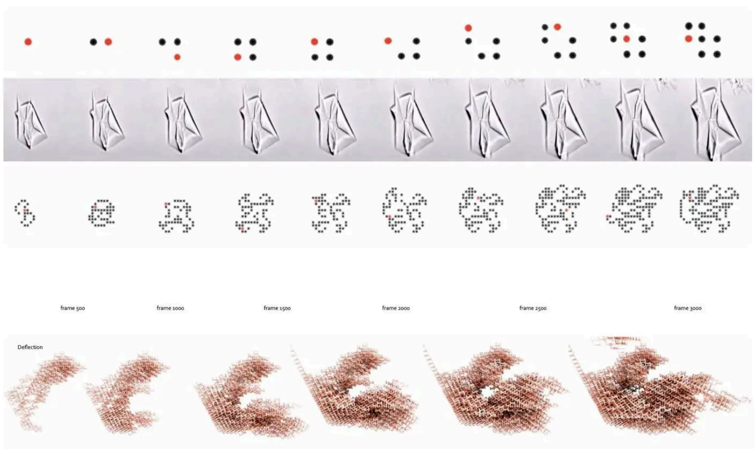

Fig. 1 A CA "seed" may generate a large set of outcomes by activating the discrete grid state from a simple on and off (0 to 1) in its basic form, controlled by predefined rules, such as solar radiation predictions, suitable for context with low data resolutions.

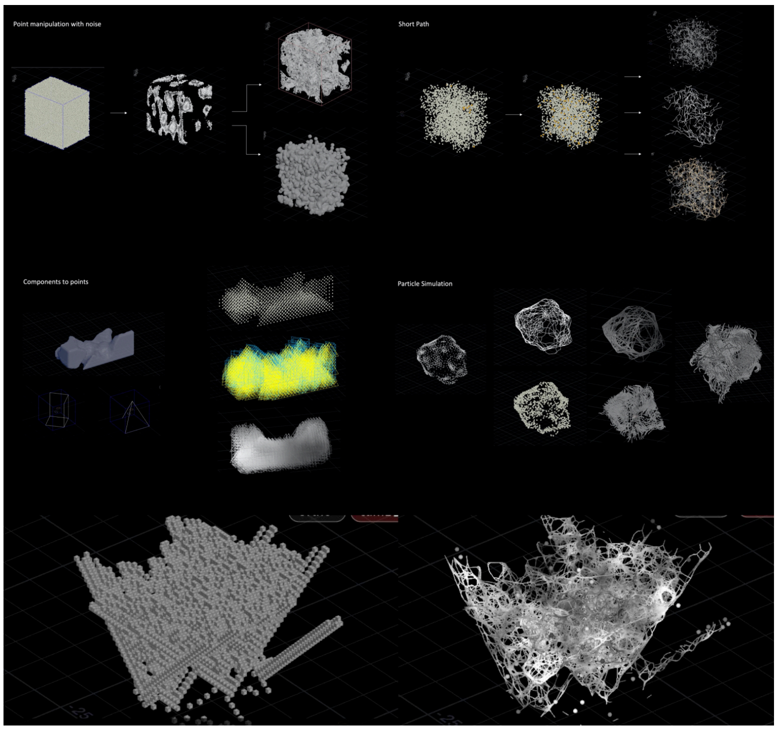

Fig. 2 The 3D data flow creates a feedback loop between Rhino and Houdini using discrete aggregation logic of CA, with voxel as proxy to define the geometry growth.

4 Design Outcomes

These projects hope to demonstrate how the same pipeline can lead to diverse design output with its combinatorial quality. The two projects have different contexts and local organisms as inspiration: one in the northern hemisphere, one in the southern hemisphere. Although both contexts are relatively close to the equator, they suffer from different climatic concerns: desert and highland climates. Both projects tried to combine active and passive solar strategies into one design as the surrounding environments are relatively scarce in resources due to socioeconomic or geographical reasons.

The Sanna Tower is a low-rise refugee shelter design situated in low-income fabrics, next to Alawi Mosque, Sharmah, Yemen. The project feedback to local society by enabling self-sufficient energy structures. As a seaport country, the design took inspiration from commonly found waste materials—discarded shipping containers that can be assembled and disassembled easily as temporary shelter structures—and upcycle them with PV units into energy resilient refugee blocks. The challenge with Modular Integrated Constructions (MiC) is to define a distribution logic that utilizes the site boundary effectively without homogenising the landscape. Participants learnt from the aggregation strategies of Marine Sponge: the organisms stack upon one another without being in direct competition for light. Here, sunlight instances became an instructor of CA distribution.

The coast of the Red Sea (Tihamah) has a desert climate, generally hot but suffers from daily temperature differences that range from 15-35°C in inland areas. Being relatively close to the equator, daylight often comes from most directions all year round.

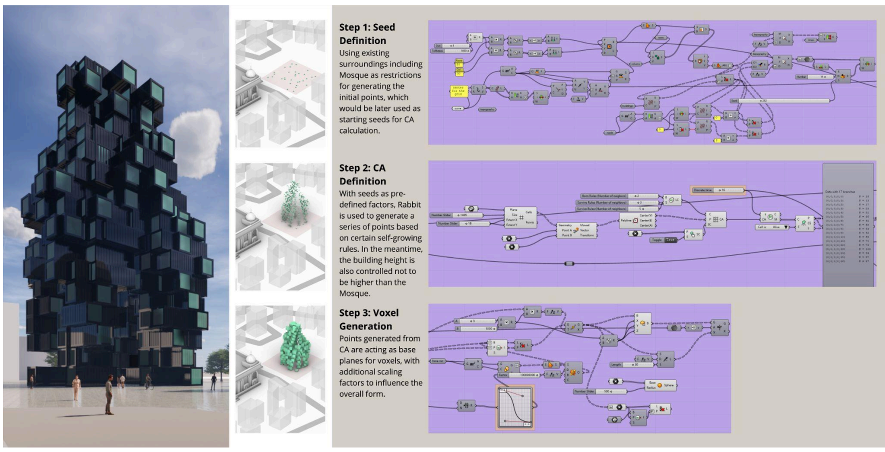

Fig. 3 The Sanna Refugee Tower. Participants credits: Chowon Kang, Yutong Zhang.

The design utilised a combination of active and passive solar strategy by playing with the orientation of units. Participants considered installing windows made with Luminescent Solar Concentrators (LSCs), which are ultra low-cost plastic doped with fluorescent dyes that harvest light while keeping windows semi-transparent.

The design process began with solar simulation to structure a voxel grid with CA, then upsampled the voxel grid with container models. The same process was carried out over a number of sites in adjacent areas to find out which site is the best for light-harvesting. Finally, participants chose this Mosque, which is a cultural site for solidarity, religious services and charity work, and the area has shown to have more Mosques than supermarkets. The flexibility allows for the same workflow to be applied to each of those sites to assess possibilities of temporary towers.

The L1ch3nic Sph3ribl0b5 is a project designed for hikers on the highland of Markawasi Plateau Geopark, Lima, Peru. The hiking tracks are in far locations, the project offers resting spots which would provide water, electricity, and light for the travellers. Participants speculate on micro-spherical solar cells and vapour-harvesting design for a combination of active and passive strategies.

Spherical solar cells are an emerging technique that tries to tackle daylight direction problems. As sunlight is not uniform in the natural environment and the sun is always moving, the typology of the sphere helps to catch light coming from all directions, ensuring at least 50% of the overall surface is in contact with light at all times, even catching those reflected and refracted from clouds and water. The design also includes a structural component and a web that catches and condenses vapour from the temperature change at dawn to provide water.

Fig. 4 The LICHENIC SPHERI-BLOBS. Participants credits: Carlos Rivera, Manuel Halim, Gao Xiang, and Mason Mo.

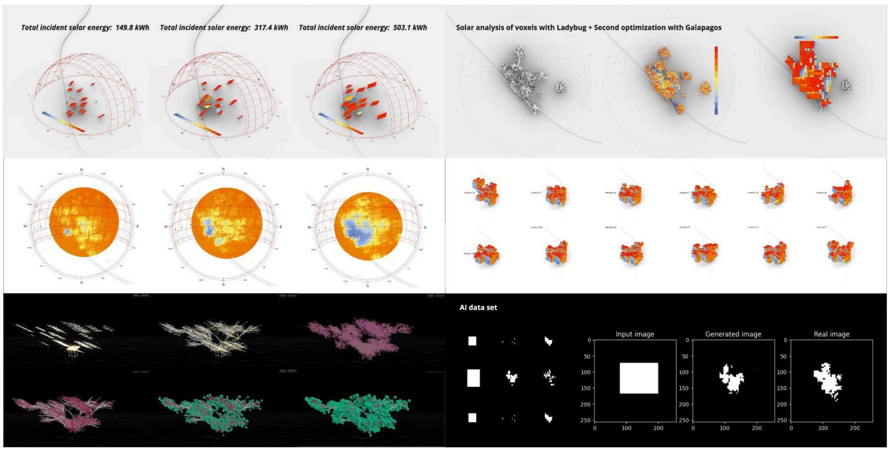

Fig. 5 Learning from Thallus Lichen that is often found in the highlands and able to collect moisture in the air, Participants simulated the growth pattern in houdini, which were then discretized into voxels. Afterwards, they trained the ML pix2pix with solar data, and the output in pixel format was then stacked into voxels. Finally, the second set of voxels output were imported back in houdini to transform the growth pattern.

5. Comparing Computational Performance

Comparing CA as the teacher and GAN as the student, they outperformed each other at different tasks. CA for optimization relied on population-based methods, utilizing selection, mutation, and crossover to refine candidate solutions. These algorithms directly optimize design parameters or structures by evaluating their fitness in Rhino3D. These simulations were computationally expensive, requiring substantial time and resources, with evaluations taking hours to days for a thousand generations. They did not require pre-existing datasets and are suitable for contexts with low-resolution data (e.g. a single screenshot, satellite images); they did, however, need significant data storage for simulation results and population data. Despite the high computational costs, CA maintained a diverse population, allowing for multi-objective optimization and producing high-fidelity solutions based on physics simulations.

GANs operated on the core principle of adversarial training between generator and discriminator networks. They were designed to learn and generate realistic data distributions, which could be adapted to generate optimized designs very quickly when trained (within seconds vs the hours needed by CA). However, GANs require datasets of images or structures for training, and are often difficult to apply to low-resolution targets, such as those sites chosen by participants of our workshop. The algorithmic quality is highly dependent on the dataset's size and diversity, thus, they make great students for CA, which could evolve solutions from very little data but takes a long time to process. Although GANs training could be computationally intensive, often taking hours on GPUs, once completed, inference was remarkably fast (within seconds). Further, it required much less storage capacity compared to CA.

Overall, CA were best suited for direct optimization problems with explicit fitness functions, while GANs excelled in generative modeling and rapid design synthesis post-training. CA had straightforward evolutionary steps, but the evaluations were costly, whereas GANs involved complex adversarial training. When combined for hybrid approaches, design iterations could leverage the strengths of both methods, balancing quality with time costs.

Table 2 Comparing computational performances of CA as the teacher and GAN as the student.

Aspect	**CA**	**GAN**
Iterations/ Generations	900–1000 evolutions for convergence.	600 training iterations, 500 samples, 256x256 px each.
Compute Time	6–7 hours for 900–1000 generations.	8–10 hours on GPU. After training = ~instantaneous.
Data Storage	70 GB for 900–1000 evolutions.	2–4 GB (200 MB checkpoints × 12 saves over 600 iterations).
Intermediate Data	High (simulation outputs, population states).	Moderate (gradient updates, logs).
Hardware	CPU (parallelization limited by simulation bottlenecks).	GPU-accelerated (training); CPU/GPU for inference.
Scalability	Time/data grow linearly with generations.	Storage scales with checkpoints; training time depends on the dataset.
Use Case Suitability	Context with little data; generating synthetic data	Data-rich context; almost instantaneous inference.

6. Human-Machine Collaboration

Although the pipeline has been shared in the same manner, each team of participants deployed it with variability. The most common workflow can be summarised as follows.

After preliminary contextual and bio-inspired research for concept framing, designers started the algorithmic process in three ways: (a) a parametrically defined form, (b) a set of manipulatable data points, or (c) solar simulation to structure a voxel grid.

The main challenge was how the complexity of bio-inspired form can be processed by solar analysis and CA with the least amount of computing power and run time so the pipeline can be democratised amongst team members with varying computing capacity.

Two solutions came up. The first was demonstrated by LICHENIC SPHERI-BLOBS: asking the CA to teach ML, which automated tasks in an instantaneous manner when trained. The second was tested by The Sanna Tower: separating the generation of complex geometry and instruction of unit aggregation. The former can be done with (a) & (b), whereas the latter are done with (c) as geometry proxies: the voxels are taken as empty data vessels that can be upsampled with complex geometries in Houdini.

In this way, the computing power and time are largely minimised because CA were only processing a set of boxes, although it could compromise the accuracy on a micro-level, the evaluation on a macro-scale remained relatively the same. The key here is to balance the resolution of the discrete grid over the predefined volume. However, it could raise issues of continuity between discrete units in later design phases, which has to be solved through designer intents. There were generally two approaches to tackle it: discretise a continuous shape or aggregate discrete units with connectors, which the selected projects had illustrated.

The overall design process took 3-5 days.

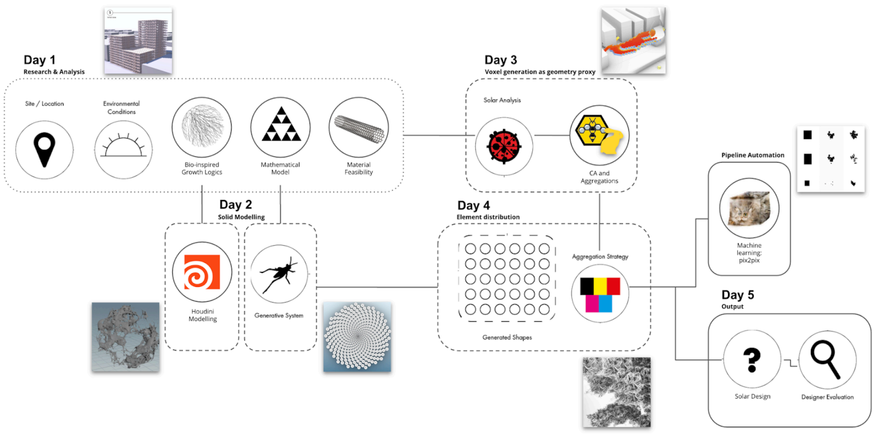

Fig. 5 The proposed hybrid workflow combining CA, ML, and Houdini growth algorithms to balance between computational efficiency and designer needs.

The tensions and opportunities negotiated through this technological framework was critically captured by participants' reflections:

> "Through this iterative design workflow, a proposal is able to adapt to different conditions in the same environment, and opens up possibilities for different alternative designs. **We architects base our design skills and criterias on personal learning experiences. AI does pretty much the same, with the difference that it has the capacity of choosing combinatorial paths of data** that us, human beings, wouldn't even be capable of thinking of. AI allows us to see things that we are not able to see at first sight"."

> "Through this workshop pipeline, a symbiotic relationship between human and machine has been created. **Decisions made by the architect, evaluated by machine, getting a product, and going back again in the loop.** A question is posed for the future. What exactly is the architect's role in such a design process?"

> "The future of design relies on self-configuring architecture. With all the design extension tools that architects have got at the moment, **we can make such accurate proposals that can adapt to the slightest environmental changes; per month, per day, per hour, per minute.** This fact clashes with the reality of today's architectural products; the static, the perpetual."

On this note, the collaboration was not simply human-machine, but also machine-machine and human-human, without whom, the diverse understanding of the experiment would not have been possible.

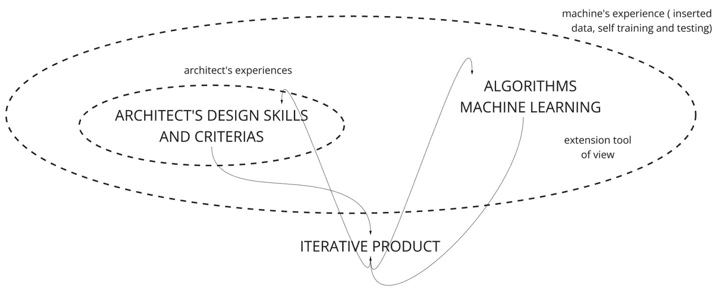

Fig. 6 Participants' diagrammatic reflection on human-machine collaborative workflows.

7. Discussions & Conclusions

By exploring the role of CA as pedagogical agents to translate contextual intelligence into trainable protocols for ML, the study contributed to three critical gaps in architectural computation: 1) the contextual void in AI-driven design, where prior approaches prioritized stylistic generation over environmental-cultural synthesis; 2) the scalability limits of rule-based systems, overcoming compute bottlenecks through ML acceleration; 3) the biomimicry implementation gap, moving beyond aesthetic metaphors to elements distribution logic.

The outcomes highlight a pedagogical paradigm where synthetic intelligence learns from each other into self-advancement, guided by human priorities and needs. The experiemnts tried to navigate the complexity of implementing symbolic automata with connectist ML by treating contextual information as a scaffold for neural networks. By doing so, it also tests the boundaries of translating between discrete computations' rigor with organic design expressions.

However, several limitations must be acknowledged, including the closed-loop feedback between synthetic datasets, which indicated that real-world performance validation remained a future endeavour. Additionally, cultural adaptation relied heavily on designer intervention, indicating a careful design of engagement protocols with decision nodes alongside computational automation procedures. Finally, there is an opportunity to align discrete and voxel based computation with BIM or supplier databases to enhance implementability of the designs.

As sustainable architecture confronts escalating climatic and socioeconomic complexity, this research argues for machines that learn contextually—not autonomously—as students of ecological wisdom and a human-steered interpreter of the world. The findings and limitations pointed to areas for further research in the quest for a more site-sensitive and design-adaptive computational process.

Acknowledgement. We sincerely thank all workshop participants for their contribution, especially Chowon Kang, Yutong Zhang, Carlos Rivera, Manuel Halim, Gao Xiang, and Mason Mo; and Digital Futures for facilitating a workshop platform.

References

1. Bolojan, D., et al. (2023). Latent Design Spaces: Interconnected Deep Learning Models for Expanding the Architectural Search Space. In Architecture and Design for Industry 4.0 (pp. 201-223). Cham: Springer International Publishing
2. Chaillou, S. (2020). Archigan: Artificial intelligence x architecture. In Architectural intelligence: CDRF 2019 (pp. 117-127). Singapore: Springer Nature Singapore.
3. Charitonidou, M. (2022). Towards a civic approach to urban data: The myths of digital universalism. Computing, 20(2), 238-253.
4. Ertan, İ. B., & Adem, P. Ç. (2024). Computational Design with Kurtboğaz: The Generation of Timber Structures with an Aggregative Design Algorithm. Journal of Computational Design, 5(2), 235-258.
5. Frazer, J. (2002). A natural model for architecture: The nature of the evolutionary model 1995. Cyber Reader Critical Writings for the Digital Era, 246-255.
6. Koh, I. (2023). Architectural and Social 'Word Forms' Slippage with Deep Learning. In CAAD F (pp. 112-125). Cham: Springer Nature Switzerland.
7. Köhler, D. (2017). Large City Architecture: the mereological mode of the quantified city. Journal of Parallel, Emergent and Distributed Systems, 32(sup1), S163-S172.
8. Ng, P., et al. (2021). AI In+ form: Intelligence and Aggregation for Solar Designs.
9. Ng, P., et al. (2022). An Info-Biological Theory Approach to Computer-Aided Architectural Design. In CAADFutures (pp. 436-455). Singapore: Springer Singapore.
10. Retsin, G. (Ed.). (2019). Discrete: reappraising the digital in architecture. Wiley.
11. Wolfram, S. (2005). Cellular automata. Modeling Chemical Systems.