REINFORCEMENT LEARNING FOR ARCHITECTURAL DESIGN-BUILD

Opportunity of Machine Learning in a Material-informed Circular Design Strategy

CHIEN-HUA HUANG
[1]*Institute of Architecture, University of Applied Arts Vienna*
[1]*chienhua.huang91@gmail.com*

Abstract. This paper discusses the potentials of reinforcement learning in game engine for design, implementation, and construction of architecture. It inaugurates a new design tool that promotes a material-informed design-build workflow for architectural design and construction industries that achieves a comprehensive circular economy. As a proof of concept, it uses the project "Reform Standard", a machine-learning-based searching system that designs new shell structures composed of existing wasted materials, as a demonstration to discuss how reinforcement learning, machine vision and automated searching algorithm in the game engine can promote a material-aware design and converts wastes into construction materials. The demonstrator project sorts and transforms irregular chunks of wasted broken plastics into a new form. Instead of recycling those wastes in an energy-intensive process, the game engine is capable of finding the intricacy and new machine-oriented aesthetics in those otherwise neglected wastes. Furthermore, future research directions such as robotic-aided construction are discussed by exposing the potentials and problems in the demonstrated project. Finally, the future circular strategy is discussed beyond the demonstrated tests and local uses. The standardization of material, legislation and material lifecycle needs to be comprehensively considered and designed by architects and designers during conceptual design phase.

Keywords. Reinforcement Learning; ML-Agents; Unity3D; circular design; geometric analysis.

1. Introduction

Machine learning (ML) has been rapidly advancing and becoming accessible for design practice. Plugin packages such as RunwayML for web-based platforms, ML-Agents kit for Unity, and Owl for Grasshopper allow designers to utilize computational power for computational design and modeling. Examples of applications of artificial intelligence (AI) and ML in the architectural field have recently emerged. ArchiGAN, for instance, is a generational tool based on

PROJECTIONS, Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2021, Volume 1, 171-180.

supervised learning algorithms for generating building plans (Chaillou, 2020). ML is now also being used in the engineering field to optimize structural configuration (Aksoez and Preisinger, 2020). Reinforcement learning (RL), a subcategory of ML, is also increasingly being explored in the design industry due to its interactive characteristic. RL, as defined by Howley and Mousavi (2018), 'involve[s] the strategy of learning via interacting (sequences of actions, observations, and rewards) with the environment' (p. 426). ML not only enhances efficiency and improves effectiveness but also opens up opportunities for innovative design practice.

In computational design, it implies that RL can play an increasingly significant role in material-informed design due to its ability to shift 'from a mere instrument of production to an agent of heuristic advancement' (Witt, 2016, p. 115). Hence, it may be able to build a comprehensively circular design strategy to reinforce the 'cradle-to-cradle' strategy, which perceives by-products and wastes as resources for other products (McDonough, 2002). Specifically, design can start from thinking of the whole design-build cycle and taking into account the massive fragments of waste and materials. Hence, architects' design criteria and responsibility can be expanded toward the full perspective of the circular strategy. Several works have demonstrated the need for and possibility of designing with waste and non-standard materials, such as in projects like Blob Wall by Greg Lynn, Mind the Scrap by Certain Measures, and Branch Formations by Conceptual Joining. However, these examples barely expand beyond its very specific function and styles. The challenge is caused by the computational-consuming analysis and design of complex geometries, which is time-consuming for designers. Instead of implementing an additional energy-intensive recycling strategy to reduce construction waste, a new ML-driven design workflow that directly reuses waste materials with irregular geometries as design inputs can significantly reduce waste production and potentially inaugurate new machine-oriented aesthetics.

This paper presents such new workflow, which takes advantage of RL for a material-informed design method to achieve a comprehensive circular strategy. The research in this paper discusses the potentials of RL in game engines for the design, implementation, and construction of an architecture that contributes to a comprehensive zero-waste design-build strategy and feasible construction affordance are discussed. A plastic structural shell is used as a demonstrator project to illustrate the potentials, problems, and methods involved. The paper concludes with the potential, limitations, and further research directions of RL/ML and the material-informed design strategy to address their application in creative design and production, thus providing diverse design outputs that take into consideration environmental concerns.

2. Methodology and Fabrication

2.1. OVERVIEW

The demonstrator project is an experiment on the design and assembly of a structural shell composed of wasted broken plastic chips. A method developed with multiple platforms, such as Rhinoceros, Grasshopper, Unity, and Reality

Capture, was applied to set up the comprehensive workflow. It consisted of inputs, a digital RL-based solution-searching program for assembling structural shells and outputs of instructions for the construction of each fragment and for the overall construction (Figure 1). In this paper, the digital workflow and ML-Agents, a Unity plugin for training intelligent agents via RL (Juliani et al., 2018), are presented in this project to showcase the power of RL in finding the shell assembly with the best structural performance, and thus to showcase its potential in promoting a material-informed circular strategy.

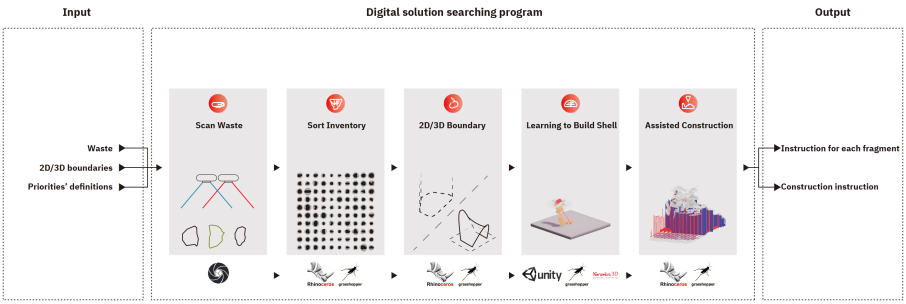

Figure 1. Diagram illustrating the program workflow.

2.2. SCAN WASTE

The first digital workflow process requires architects' first input sets of waste geometries, which are digitized with RealityCapture (RC) so they can be analyzed in Rhinoceros and Grasshopper. Figure 2 shows the setup for photogrammetrically scanning the target with the highest possible resolution to build a precise 3D model resembling its physical counterpart. High-resolution scanning can ensure proper geometrical analysis and digital assembly (Figure 3).

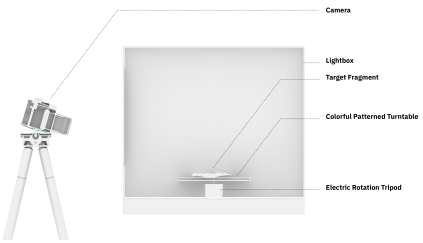

Figure 2. Setup for photogrammetry.

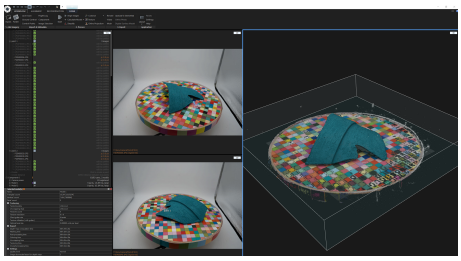

Figure 3. RealityCapture screenshot showing the photo samples and the digital model of one plastic fragment built via photogrammetry.

2.3. SORT THE INVENTORY

In the second step, each geometry is analyzed and sorted to enable sequential assembly in the searching process as the Unity program can place only one fragment at a time. Through Grasshopper, different data are extracted from each fragment for their implementation in future processes (Figure 4). Specifically, projection areas are used to define the maximum boundaries of the assembling procedure. The surface areas, HU invariants, number of control points, and erosion maps‘ areas are multiplied with different weights to sum them up into one number for every fragment so as to sequence the inventory. The skeleton’s midpoint and end points are exported to Unity so that ML-Agents can act according to the respective geometries’ topologies. To sum up, such data mitigate the burden on RL so that Unity does not need to encapsulate the computation-consuming sorting strategy.

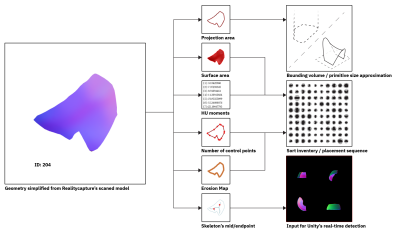

Figure 4. Diagram illustrating the types of data analyzed and their implementation in the following processes.

2.4. DEFINE THE 2D / 3D BOUNDARY

A 2D perimeter or a 3D volume is defined as a boundary for constraining the fragments‘ positions during the learning process. The boundary can be directly inputted by the user as a closed polyline or a closed brep. In the first case, unlimited height is allowed for the assembly within the regions. As for the 3D bounding geometry, the program resembles the solution of bin-packing problems with random geometries. The boundary is set to control the possibility of assembly

within a reasonable range. If the boundary is notably larger than the sums of all the fragments' projection areas or volumes, the efficiency of RL will remarkably diminish. Therefore, the boundary resembles the site perimeter and regulates the overall forms, which helps control design and avoid unnecessary learning time.

2.5. MACHINE LEARNING SEARCHING

After the above-mentioned conditional setup, all the sorted fragments and optimized boundaries are inputted into Unity for an assembly simulation test to search for the best structural shell. The most important benefit of introducing RL in ML-Agents is that the complex methodology of constructing a structural shell with various criteria does not need to be exclusively developed by architects. Rather, designers merely input priority criteria to direct ML-Agents. ML-Agents summarizes the data collected from the previous steps and learns to build a method that can construct a shell with different priorities defined by the users. Eventually, the method can be saved as a neural network model (NN-model), a model resembling a brain, to control the agents' behavior (Arthur et al., 2018).

In the Unity scene, each episode loads the fragment in sequence from the sorted inventory, finding a position to assemble the fragment onto the ground or other fragments, and developing a way to construct a structural shell through the observation of its environment (Figure 5). The game starts with the agent (controlled by AI) moving and rotating the fragment to find the best position based on its observation. The fragment is allowed to be assembled only if it fits all the criteria for a joint resembling a real physical connection (Figure 6). After the assembly, positive and negative rewards are assigned according to the evaluation results of the structure, height, and other performances defined by the users. Then the agent evaluates all the rewards and observation data to adjust the method for the next actions. Six major types of data are observed to train ML-Agents to assemble according to the performance of such priorities (Figure 7). These data are obtained via the cross-platform synchronization of shell model and analysis. Among the six types of observation data, the structural stability is evaluated via Karamba analysis in Grasshopper. Rhinoceros and Grasshopper are connected to Unity through a user datagram protocol (UDP) connection. The geometries are streamed from Unity to Grasshopper for Karamba structural analysis. Karamba is a Grasshopper plugin for finite element analysis that can provide real-time feedback. The analysis result is then streamed back to Unity through UDP for rewards stating. Other observation data, such as floor areas and thermal dynamics, are obtained using a camera within the game and RGB distribution analysis.

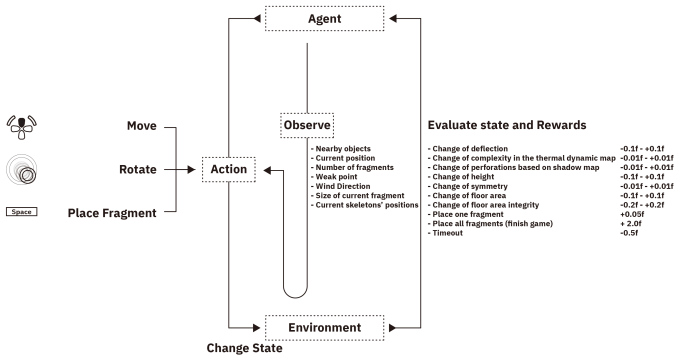

Figure 5. Diagram illustrating the processes, control, observation, and reward setup of the game for ML-Agents to learn assembly.

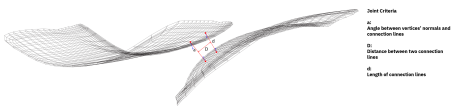

Figure 6. Diagram showing the conditions of the connection of two fragments.

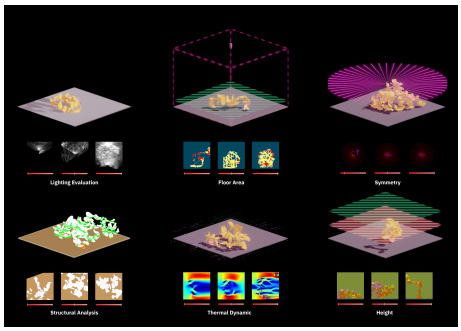

Figure 7. Diagram showing the maps and axonometric models of Unity observing the lighting variety (top left), floor area (top middle), symmetry (top right), structural stability (bottom left), thermal dynamic variety (bottom middle), and height (bottom right).

After the assembly simulation, the model of all the finished shells, their six scores, and an NN-model are saved. One shell can be selected to proceed based on the scores and the users' choices. The multiple reward evaluations and weights open up options for the users to curate their priorities according to their programs' needs (Figure 8). However, the nature of the evaluation may bring about results different from the users' expectations. The problem is caused by the different non-linear training trajectories used to obtain better scores for different evaluations. For instance, structural stability is much harder to achieve than area coverage as the former requires more data observations, detections, and

explorations. In the demonstrator project, a shell with the best structural stability was selected to proceed to the construction phase.

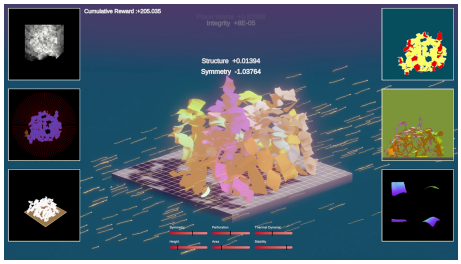

Figure 8. Unity screenshot of a training session.

2.6. ASSISTED CONSTRUCTION

After a decision is made by the user and the machine, the workflow will generate guidance on how to treat each fragment. This will aid the assembly of each fragment and ensure control and precise sequence. First, the connection points, drilling points, and their connection IDs indicating the fragments to be jointed are projected and marked on the geometry to exclude the digital devices' dependency (Figure 9). Plastic blind rivets were chosen as the joint because they are efficient for assembly purposes, homogeneous to plastic fragments, and lightweight (Figure 10 and 11). Despite the construction simplification, however, unexpected difficulties may arise when the assembly sequence is changed due to the need for temporary support. Thus, the whole procedure requires a fixed sequence.

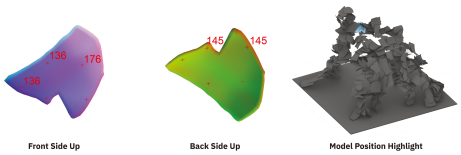

Figure 9. Rhinoceros screenshot of individual instructions for a sample fragment (left and middle) and its position in the model (right).

Figure 10. Under construction: Projection and marking (left), drilling (middle), and connecting (right).

Figure 11. Under construction: assembling and labeling.

3. Discussion and Future Researches

The successful design-build experiments on the workflow and the workflow's geometrical analysis, structural analysis, RL searching, and assembly prove the feasibility of further applying this process to other structural design practices. Figure 12 presents the final model in the demonstrator project. The design of the model subsequently prioritizes structural stability, floor area, and height among the six given types of observation data. The model was efficiently assembled even with only manual labor. However, several deficiencies occurred in the workflow experiment presented herein due to resource and technological limitations. First, the structure had around 2% deviations in width and length and an around 10% deviation in height from the digital model. This was caused by the imprecision of manual assembly, the unmatched material quality between the Karamba analysis results and the plastic fragments, and the different gap widths between Unity and the structural model. Second, not all the actions in the RL implementation can be rationally explained due to the nature of trial-and-error solution searching. Thus, RL can provide only solutions that seem intuitive, in which the reasons for some actions cannot be given. For example, in the final model, the columns have various types of composition, but the efficiency and stability of such columns cannot be compared within the RL framework. Nevertheless, RL implementation offers powerful insights and possibilities for architects that can be further optimized and edited.

Figure 12. Result: The model's front-perspective view.

To further expand the realm of the circular design strategy with materials as inputs, the future researches can expand the application of the workflow through three major aspects: robotic fabrication to replace manual work, introduction of the composition of mixed materials, and an online inventory to enable efficient mass collaboration and fabrication.

3.1. ARTISANAL ASSEMBLY VIA ML AND ROBOT

By combining ML and robotic assembly, architects can adopt an individual artisanal assembly approach. In the demonstrator project, the workflow outputs instructions for complicated manual construction, but robotic arms have the potential to handle such a complex assembly. Additionally, RL can enable robotic behavior and can interact with the environment (Kober et al., 2013). As such, it can be made part of the ML input so that during the design, the robotic movement can become a design parameter. Specifically, after the digital assembly phase, the program can train the robot to find an assembly strategy, proceed to physical control, and perform a task similar to digital simulation. Robotic fabrication for such an assembly can be beneficial and efficient due to robots' capacity for cooperation and possible involvement in the early design phase (Parascho et al., 2018). Therefore, future experiments with robotic assembly in the digital environment will ensure the profitability of the workflow and will contribute to the industrialization of this procedure.

3.2. MIXES OF MATERIALS AND JOINTS

The mixed assembly of different materials is feasible by allowing the use of other types of joints, which offers a wider variety of building types. The double-pin joints in the demonstrator project can already be applied to a significant number of materials, such as metal sheets and planks. To come up with a complex structure, a mix of different joint types may be necessary to enable other structural systems. Hence, future researches should investigate the method and reward systems in RL to search for multi-joint solutions.

3.3. MATERIAL AND COMPUTATION E-CROWDSOURCING

The digital production of the design algorithm promotes a potential new mass collaboration method through e-crowdsourcing. The potential new ways of practice through e-crowdsourcing are significantly beneficial as "the ideas of permanent variability, parametric mass customization, and digitally driven mass collaboration that designers test drove during the age of the first digital turn are now spreading in all areas of contemporary society, economy, and politics" (Carpo, 2017, p. 75). The inventory of fragments for design inputs can ideally be scanned, stored, shared, and delivered everywhere. As such, the users do not need to scan all the objects found. Rather, they can order the wanted construction materials from the web inventory and build a desired geometry from the selection. This will eventually help the users build more comprehensive structures with various textures and programs, which can help build a more comprehensive inventory that considers all kinds of non-standard resources.

4. Conclusion

When the difficulties of fitting geometry to design are dissolved by shifting the design cycle and applying RL searching, significant opportunities are opened up in a new realm of practice and design, especially the possible circular strategy. The nature of the workflow reinforces designers' responsibility to take into account the complex material reality in the environment. "Reform Standard" demonstrates how architectural design can implement ML in a design−build cycle and prioritize the material inputs. Despite the current limitation of computation power and RL's nature of unexplainable rationale, the workflow can effectively and smartly assemble complex geometries whose massive data are beyond a human designer's capacity to understand. Various future research potentials are unfolded by this workflow, which makes the sustainable design more industrially applicable, intelligent, and efficient. "Reform Standard" thus illustrates that RL has the potential to give rise to a material-informed design method and to fully sustainable building practices.

Acknowledgements

This research was based on a master thesis titled "Reform Standard" completed in June 2020 at Studio Greg Lynn, Institute of Architecture at the University of Applied Arts Vienna (die Angewandte), under the supervision of Prof. Greg Lynn and with the assistance of Bence Pap, Kaiho Yu, Maja Ozvaldic, and Martin Murero. Special thanks are extended to Lisa-Marie Androsch, Martin Lai, Wing Yan Joyce Lee, Yi Jiang, Yu Zhao, and Zach Beale.

References

Aksoez, Z. and Preisinger, C. 2020, An Interactive Structural Optimization of Space Frame Structures Using Machine Learning, *in* C. Gengnagel, O. Baverel, J. Burry, M. Ramsgaard Thomsen and S. Weinzierl (eds.), *Impact: Design With All Senses*, Springer, Cham, 18-31.

Arthur, J., Berges, V.P., Vckay, E., Gao, Y., Henry, H., Mattar, M. and Lange, D.: 2018, "Unity: A General Platform for Intelligent Agents" . Available from <https://www.researchgate.net/publication/327570403_Unity_A_General_Platform_for_Intelligent_Agents>.

Carpo, M.: 2017, *The Second Digital Turn. Design beyond intelligence*, MIT Press.

Chaillou, S. 2020, ArchiGAN: Artificial Intelligence x Architecture, *in* P. Yuan, M. Xie, N. Leach, J. Yao and X. Wang (eds.), *Architectural Intelligence, Selected Papers from the 1st International Conference on Computational Design and Robotic Fabrication (CDRF 2019)*, Springer, Singapore, 117-127.

Kober, J., Bagnell, J.A. and Peter, J.: 2013, Reinforcement Learning in Robotics: A Survey, *The International Journal of Robotics Research*, **32**, 1238-1274.

McDonough, W.: 2002, *Cradle To Cradle : Remaking the Way We Make Things*, North Point Press, New York.

Mousavi, S., Schukat, M. and Howley, E.: 2018, Deep Reinforcement Learning: An Overview, *Proceedings of SAI Intelligent Systems Conference (IntelliSys) 2016*, 426-440.

Parascho, S., Kohlhammer, T., Coros, S., Gramazio, F. and Kohler, M.: 2018, Computational design of robotically assembled spatial structures, *Conference: Advances in Architectural Geometry 2018*, Gothenburg.

Witt, A.: 2016, Cartogrammic Metamorphologies; or, Enter the Rowebot, *Log Journal of Architecture*, **36**, 115-124.