Deep Reinforcement Learning for Autonomous Robotic Tensegrity (ART)

Tyson Hosmer
The Bartlett School of Architecture / UCL

Panagiotis Tigas
University of Oxford

1

1 Photograph: 2.2m Diameter Tensegrity Robot

ABSTRACT

The research presented in this paper is part of a larger body of emerging research into embedding autonomy in the built environment. We develop a framework for designing and implementing effective autonomous architecture defined by three key properties: situated and embodied agency, facilitated variation, and intelligence.We present a novel application of Deep Reinforcement Learning to learn adaptable behaviours related to autonomous mobility, self-structuring, self-balancing, and spatial reconfiguration. Architectural robotic prototypes are physically developed with principles of embodied agency and facilitated variation. Physical properties and degrees of freedom are applied as constraints in a simulated physics-based environment where our simulation models are trained to achieve multiple objectives in changing environments. This holistic and generalizable approach to aligning deep reinforcement learning with physically reconfigurable robotic assembly systems takes into account both computational design and physical fabrication.

Autonomous Robotic Tensegrity (ART) is presented as an extended case study project for developing our methodology. Our computational design system is developed in Unity3D with simulated multi-physics and deep reinforcement learning using Unity's ML-agents framework. Topological rules of tensegrity are applied to develop assemblies with actuated tensile members. Single units and assemblies are trained for a series of policies using reinforcement learning in single-agent and multi-agent setups. Physical robotic prototypes are built and actuated to test simulated results.

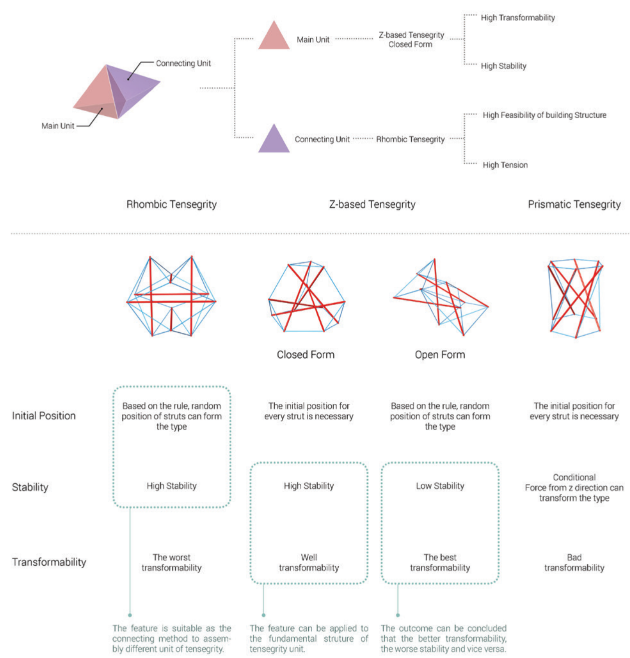
INTRODUCTION

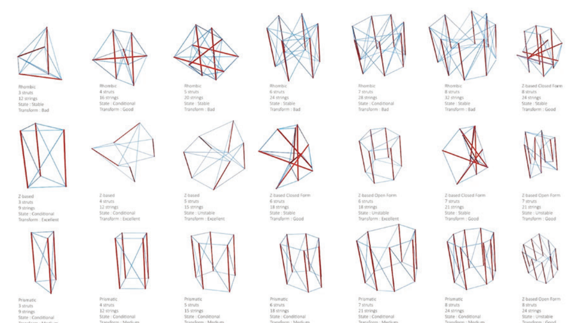
Buildings have enormous costs, energy consumption, the potential for errors, require many years to build, yet become obsolete before they are even completed because they are planned with linear life cycles consisting of raw material extraction; manufacturing; construction; operation; demolition; and disposal (GlobalData, n.d.; Ngwepe and Aigbavboa 2015). Furthermore, most buildings are considered as a series of layers with independent life cycles or paces, named by Frank Duffy as six shearing layers: Site, Structure, Skin, Services, Space Plan, and Stuff (Brand 1995).

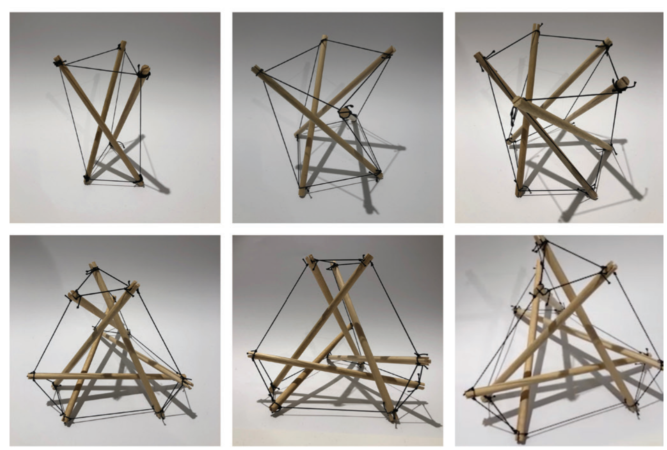
We typically focus on optimising solutions addressing individual segments of the linear life cycle rather than reappraising the cycle holistically. The post-war rise of computation and manufacturing pushed architects toward automation of manufacturing through prefabrication, yet buildings require degrees of localised specificity, flexibility, and variation not seen in cars and airplanes. Advances in digital fabrication in the 1990s enabled the rise of mass customisation. This has led many to approach architecture's complex requirements by designing highly variable continuous building forms, argued for as being manufactured at no added cost. These factors have led to layered, overly constrained buildings, inflexible to future change, expensive to construct and laborious to assemble.

Alternatively, living systems demonstrate extraordinary scalable efficiencies in adaptive construction with simple flexible parts. For example, nomadic ant colonies face extreme time pressure to generate foraging routes moving massive numbers of ants each day, yet through their simple parts and local rules have shown rapid efficiency in constructing adaptive bridges. The ants modulate their behaviour in response to locally changing environments to adapt to dynamic traffic conditions, recover from damage, and dissemble when underused. Efficiency is quantifiable, yet costs and benefits are not measured in single elements or separated performances, but revealed in degrees and rates of adaptation to changing environments and shifting goals (Graham et al. 2017).

The research presented in this paper is part of a larger body of work focused on embedding autonomy directly in the built environment. We establish a design method for autonomous architecture with three interrelated properties: situated and embodied agency, facilitated variation, and intelligence. This methodology is applied to Autonomous Robotic Tensegrity (ART), a case study project developing a robotically adaptive tensegrity system trained using deep reinforcement learning. It is developed through



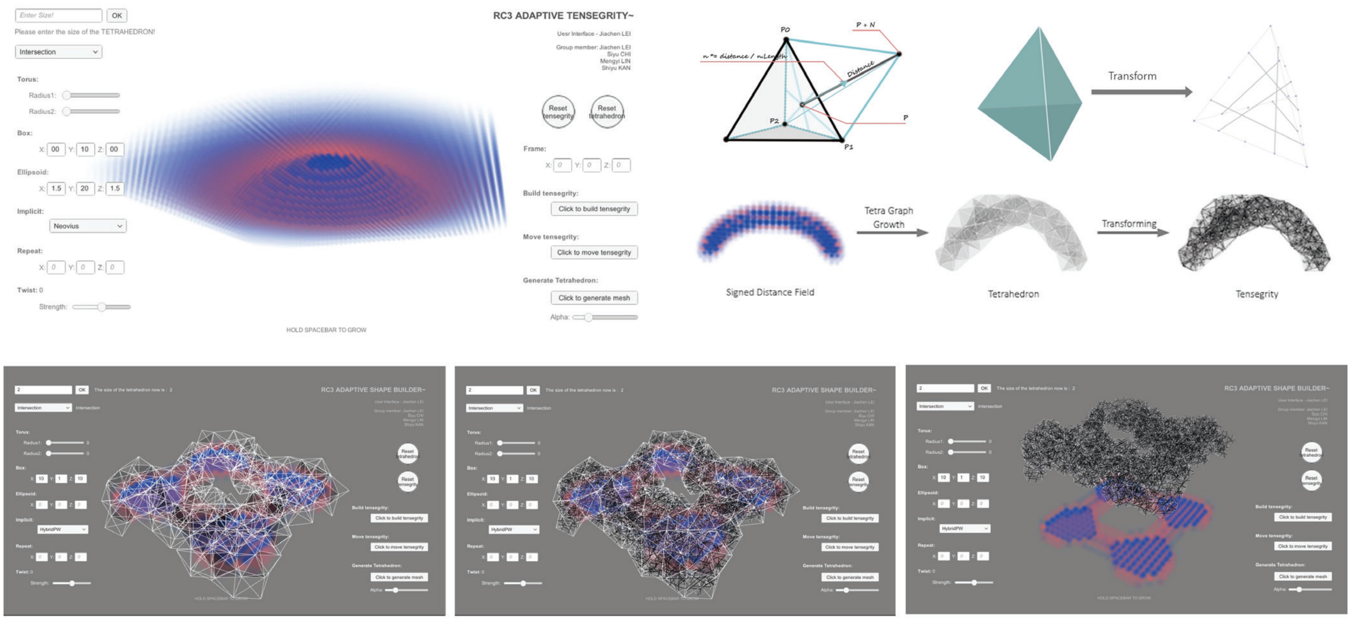




2 Tensegrity Topology Diagrams

3 Karamba based Tensegrity Behavioural Analysis

4 Photographs: Tensegrity Models



5 Images of tensegrity assembly generation process.diagrams

physical robotic prototypes, a computational framework for generating and simulating tensegrity assemblies with embodied and situated agency in a time based multiphysics environment, and multiple neural networks trained using Deep Reinforcement Learning methods, using ML-Agents Framework (Juliani et al. 2018) and Unity3D.

BACKGROUND

Automation comes from the Greek automaton meaning "self-movement", whereas autonomy comes from autonomos meaning "self law." Most major approaches to generic construction tested since the advent of prefabrication, including element assemblage, layer stratification and module composition, maintain a separation between manufacturing and assembly (Lu 2017). Typically, efficiency is argued for through the automation of design, or manufacturing, or assembly as separate unidirectional processes. Skylar Tibbitts of MIT Self Assembly Lab defines self-assembly as a process by which disordered parts build an ordered structure without humans or machine, focusing the lab's work largely on relationships between geometry and materials capable of scalable self-assembly (Tibbits 2017, 2012).

The methodology outlined in this paper focuses on moving toward autonomy by developing an interdependent relationality between the properties of facilitated variation, situated and embodied agency, and intelligence.

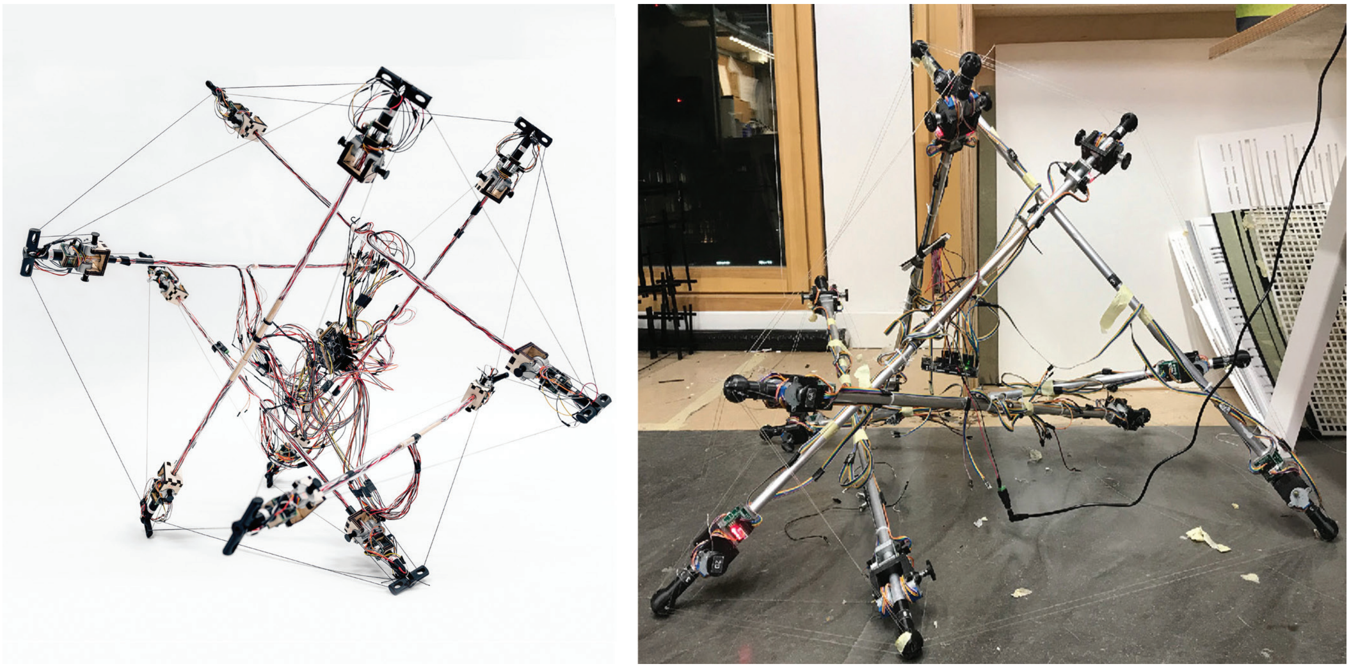
FACILITATED VARIATION

The theory of "facilitated variation" related to evolvability is the observation that the intrinsic construction of an organism, biases not only the type and amount of potential phenotypic variation, but also specifically the organism's ability to generate novel phenotypes that are useful (Parter, Kashtan, and Alon 2008).

Relating to macroscale manufacturing and assembly, MIT professor Neil Gershenfeld's concept of "digital materials" offers reversibility and reconfigurability organized by computational models that consider the combinatorics of discrete parts (Gerhart and Kirschner 2007; Popescu, Mahale, and Gershenfeld 2006). The concept of "discreteness" as a computational design principle is extended by Gilles Retsin and the Computational Design Lab at Bartlett showing potential for flexible and scalable assembly systems aligned with spatial segmentation models (Retsin and Garcia 2016; Retsin 2019).

We extend the concept of "digital materials" with the property of facilitated variation found in evolutionary biology, asserting that to enable useful future adaptation to unforeseen environmental change, requires the design of discrete building parts with effective degrees of freedom aligned with our computational model that is assembly-aware. Autonomous Robotic Tensegrity utilises tensegrity's latent potential for facilitated variation.

Tensegrity or "tensile-integrity" studied extensively by Buckminster Fuller (Fuller, 1962), consists of pre-tensioned strings and pre-compression bars. It is a reticulated self-equilibrated structural system with no bending moment



6 -7 Photographs: 1.3m Diameter Tensegrity Robots (Left to Right Icosahedral and Tetrahedral).

exhibiting many novel features, especially as a candidate for composite metamaterials (L.-Y. Zhang et al. 2018). Tensegrity structures are exceptionally rigid in relation to their small mass, and the nature of the discrete pure compression elements resting in a continuous tensioned net enables potential for effective degrees of freedom for reconfiguration.

“Biotensegrity” is a physics first approach to understanding the adaptability and organization of biological systems suggesting the flexibility and biased degrees of freedom seen in tensegrity structures is found in the behaviour of living systems. For example, forces flow primarily through our muscles and fascial structures and our bones are floating in the tension structure created by them. Biotensegrity applies tensegrity principles, geometries, and degrees of freedom in the modelling of living systems at all scales (Levin 2006; Ingber 1998).

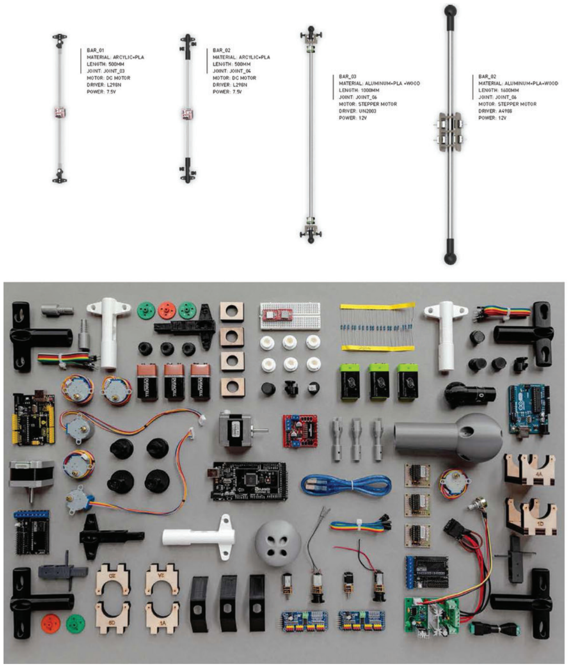
AGENCY

Agency is the capacity of an entity to act in its environment. Agent-based systems have been explored as a computational design mechanism by extending Craig Reynolds Boids algorithm for simulating collective swarm behaviour (Reynolds 1987). Two common problems persist: 1) Too many degrees of freedom result in continuous forms requiring post-rationalisation to be manufacturable, and 2) Behavioural performance is solidified in a static artifact, negating the argument of being “adaptive” to change.

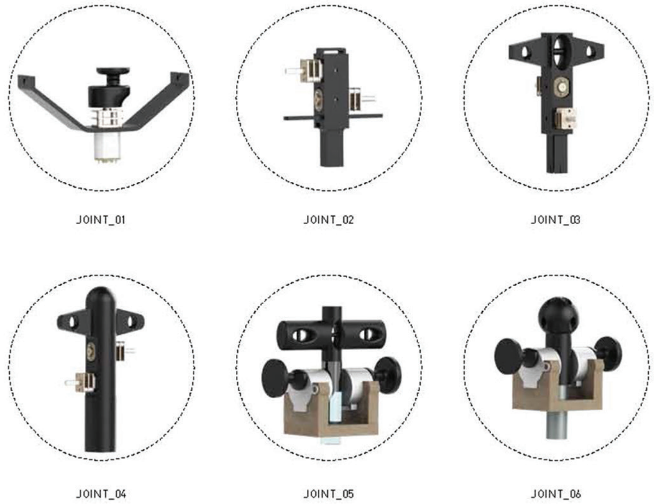
Autonomy in architecture requires agency that is both situated and embodied. Situated meaning, “situated in the world - not deal(ing) with abstract descriptions, but with the ‘here’ and ‘now’ of the environment that directly influences the behaviour of the system (Brooks 1991).” Embodied meaning that our computational model only considers relations between parts for assembly sequences that are physically feasible and translates them to the physical world.

Autonomous Robotic Tensegrity has actuators changing the length of its tension elements to rebalance the distribution of tensile forces and reconfigure the structure. Agency is situated in that the computational agent has awareness of goals and conditions through its environment. Embodiment is encapsulated in the constrained action space of our computational agents and through continuous feedback with a multi-physics simulation model calibrated through physical robotic prototypes.

Principles of swarm intelligence in nature have been applied through embodied swarm construction at Wyss Institute with Kilobots and Termes (Rubenstein, Ahler, and Nagpal 2012; K. Petersen and Nagpal 2017; K. H. Petersen, Nagpal, and Werfel 2011). Successful autonomous robotic self-assembly has been demonstrated as an extension of MIT’s “digital materials” research with BILL-E robotic platform where a relative robotic system replaces a gantry-based robotic system, enabling robots to climb on the same structure they assemble, inspect, and reconfigure (Jenett and Cheung 2017). Termes and BILL-E demonstrate scalable automation of construction and reconfiguration with



8



9

degrees of facilitated variation and situated and embodied agency, but cannot be autonomous without intelligence.

INTELLIGENCE

Autonomy can be thought of as the emergent property of an intelligent agent with sufficient degrees of freedom and facilitated variation. Russel and Norvig (Russell and Norvig 2016) define artificial intelligence as the field of building intelligent agents "which are functions taking as input tuples of percepts from the external environment, and producing behaviour on the basis of these percepts." Intelligence in that sense is the behaviour of agents "that are reacting to perceptions according to some criteria of rationality." Desired behaviour of the system is captured in objective functions. Adaptation is the change of behaviour in order to facilitate the changes of the stochastic and always-changing environment and satisfy architecture objectives. The problem of adaptability and autonomy becomes a multi-objective optimization problem of a dynamic (time-dependent) system.

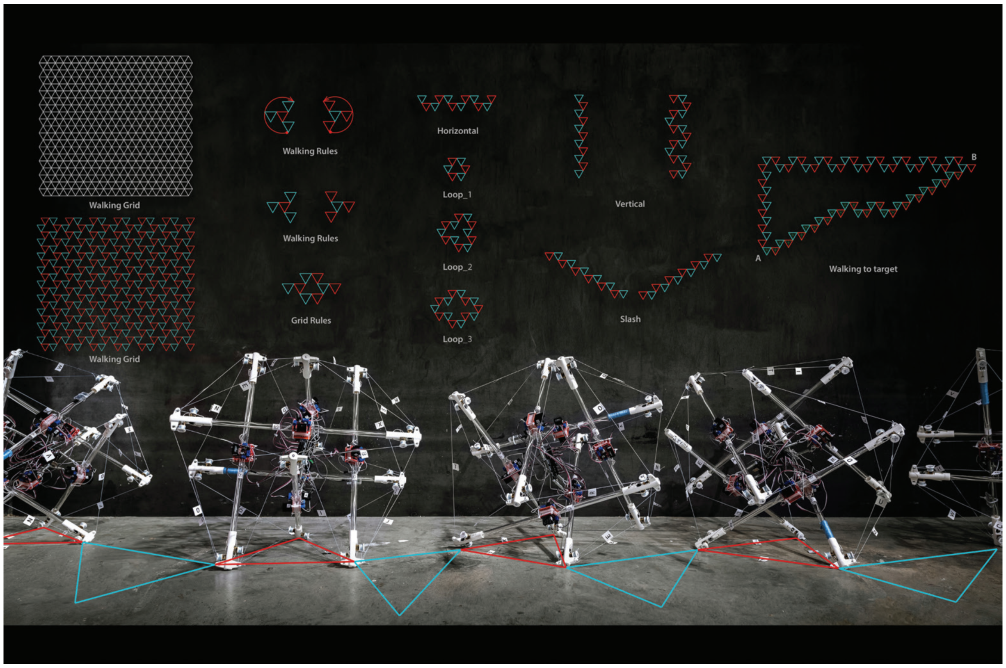
The latest wave of AI, the age of Deep Neural Networks, showed promising progress in the field of adaptive and autonomous intelligence. Deep Reinforcement Learning researchers managed to create agents that learn how to act intelligently from trial-and-error in their environment and has been applied in Robotics (Kober, Bagnell, and Peters 2013; Levine et al. 2016), Resource Management (Mao et al. 2016), Games (Mnih et al. 2013) and other fields (Li 2017). But what is the architectural relevance of Reinforcement Learning?

Reinforcement Learning (Sutton, Barto, 1998) is closely related to the field of optimal control, in which one is seeking an optimal policy for controlling a system in order to optimize an objective. However, in control theory, you have complete knowledge of the state of the system and its dynamics. In contrast, Reinforcement Learning does not assume complete knowledge of the state, but learns how to control the agent by interacting with the environment. This makes it an appealing paradigm for architecture. A building learns an adaptive policy by interacting with the environment through a situated and embodied agency while receiving feedback signals called rewards. The designer defines the architectural objectives as reward functions and trains deep neural networks to map states in actions. The state becomes a representation of the "here" and "now" of the architecture-as-agent.

The Reinforcement Learning formulation consists of the following parts, referred to as the Markov Decision Process tuple (Sutton, Barto 1998).

States: This captures the information that the agent needs to know in order to make an informed decision.
Actions: Actions are performed by the agent and have an impact on the environment they are in.
State Transitions: A function describing what is the next state, given a previous state and an action.
Reward: A function that provides immediate feedback to the agent after a state transition.

The goal of Reinforcement Learning is to find a policy (or



7

8 Photograph of robot components.

9 Diagrams / Photos of Robotic Components

10 Diagrams of walking patterns overlay on Photograph of robot mobility sequence

strategy) that maps states to actions, in order to maximize the expected reward over time. The agent learns by interacting with the environment via actions, perceiving the world via states and observations and receiving feedback at each timestep via the reward function. The designer defines the possible actions, part of what we call the facilitated variation, and shapes the behaviour of the system by designing the reward function to include the architecture objectives.

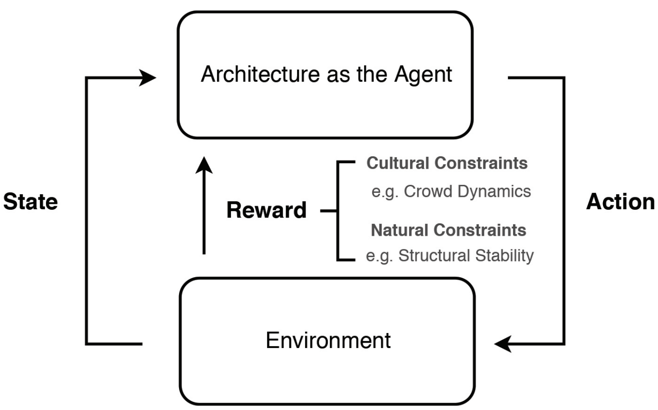


Figure 11 - Reinforcement Learning Loop

Considering architecture as a multi-objective problem, we divide its objectives into two categories: physical and cultural. Natural objectives are physical (e.g. structure, environment, comfort) and cultural objectives are socio-economic dealing with how the building is used (e.g. program, ownership, occupancy). By defining reward functions that target natural or cultural objectives, the agent learns to act to optimize these qualities.

METHODOLOGY

Autonomous Robotic Tensegrity (ART) was produced in a research through teaching context in The Bartlett Architectural Design Living Architecture Lab, Research Cluster 3 led by Tyson Hosmer, David Reeves, Octavian Gheorghiu, and Panagiotis Tigas with significant technical contributions by Ziming He and the TARSS (2017-2018) and Adaptive Tensegrity (2018-2019) research teams. The project leverages a latent potential for tensegrity structures to be dynamically adaptive by actuating the tensile members to change their length and thus continuously change the state of the structure. It has been developed both as an autonomous habitation system for Mars colonisation and as a deployable system for long span adaptive structures.

FACILITATED VARIATION: TENSEGRITY UNIT DESIGN

A large range of units were studied for potential geometric combinatorics and individual spatial and physical behaviours, geometric combinatorics, and for collective and scalable behaviour in larger assemblies. Physical behaviours were simulated with and without gravity both using Karamba for Grasshopper and in a custom simulation developed in Unity3D.

Topology design of tensegrity focuses on the possible connections between strings and bars to satisfy the tensegrity principle (L.-Y. Zhang et al. 2018). Elementary units from regular polyhedral and prismatic geometries have been researched for large scale tensegrity assembly. Tensegrity analysis is divided into four steps: topology design, self-equilibrium analysis, stability analysis, and mechanical response analysis (Zhang et al. 2018).

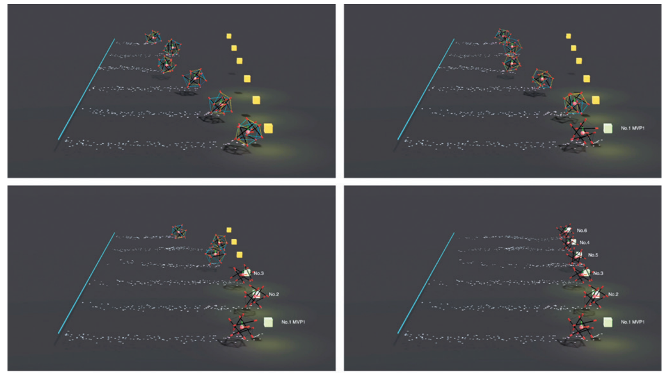
Two strategies were developed for ART's elementary units. The first considers uniform macroscale units internally inhabitable and connected into larger assemblies. Tensegrity icosahedron was selected. Its sphere-like space enables macroscale interior inhabitation and rolling-like locomotion while being rigid and reconfigurable.

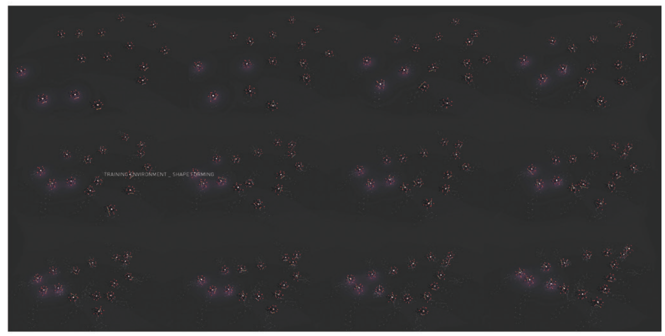
The second strategy was to develop multi-scalar non-uniform units, not individually inhabitable, but assembled into larger inhabitable spatial structures. These elementary units are derived from any tetrahedron, enabling a generalisable method of form and space generation from any tetrahedral mesh. The system is composed of two complementary types; a main unit and connecting unit. Two sub-strategies were developed, one composed of a z-based main type and modified rhombic type, while the other has a rhombic type for both the main and connecting unit. Z-based typology was found to have an ideal balance of stability and greater transformability, while the "loop" connection rule of rhombic typology provides additional stability in aggregations.

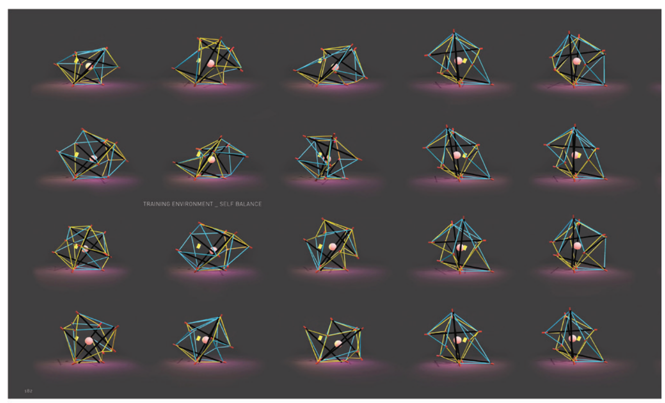
ART's computational framework includes a spatial generator for simple as well as complex nonlinear aggregations using three steps. We first define the boundaries of the assembly using a signed distance field. Next a multi-scalar tetrahedral mesh is generated through thresholds in the field. Finally, the mesh is converted using topological rules into a tensegrity structure composed of linked complementary A and B units.

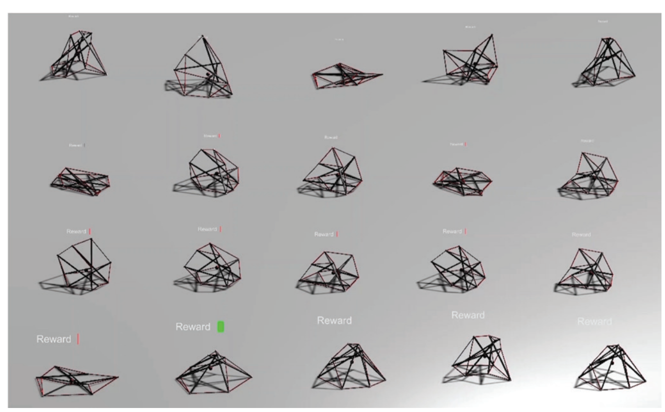
EMBODIED AGENCY: ROBOTIC PROTOTYPES

Robotic prototypes ranging from 150 mm to 2200 mm in diameter were developed and manually actuated user defined sequences for locomotion, raising the center of gravity, and dynamic balancing of multiple interconnected assemblie. Arduino actuation of DC and stepper motors was triggered through the Unity3D simulator. Nodes integrate two motors and two anchor points. Tension members are added in sequence, each connecting one motor on a starting bar to the anchor point on an ending bar. Tensile members were tested in natural fiber, nylon, and woven steel and compression members in wood, steel, and carbon fiber. Compression bars and nodes with pulleys were iteratively developed to minimize friction and integrate electronics for weight balancing.









12 Reinforcement learning simulation: Mobility Training

13 Reinforcement learning simulation: Mobility Training simulation: Collective Shape Formation

14 Reinforcement learning simulation: Agent Self-Balance Training

15 Reinforcement learning simulation: Multi-Agent Self-Erection

16 Photograph: 600mm Diameter Tensegrity Robot

SITUATED AGENCY: SIMULATOR

Elementary tensegrity units are treated as agents in our computational framework, capable of acting individually or collectively in multi-agent assemblies. A simulation environment was built in Unity3D for Mars and Earth environments, testing robustness under different gravities and ground surfaces for tensegrity structures generated by our computational model with constraints, degrees of freedom, and physical properties of our robots. Characteristics including quantity of wires actuated simultaneously, the rate of actuation, friction, and degrees of freedom were transferred from the robots to improve the accuracy of the simulator.

Compression elements are given properties of rigid bars while tensile elements are modelled as spring-like configurable joints, constrained to the end node of the compression members with free angular motion. Tension elements are actuated by changing corresponding spring rest lengths. Changing the tension in sequences enables the structure to move in different directions or change its shape. The linear spring limits were allowed a range of 0.3-2 times their initial length. Tension member length changes from the node, and tension is evenly distributed to both connected bars.

INTELLIGENCE: REINFORCEMENT LEARNING

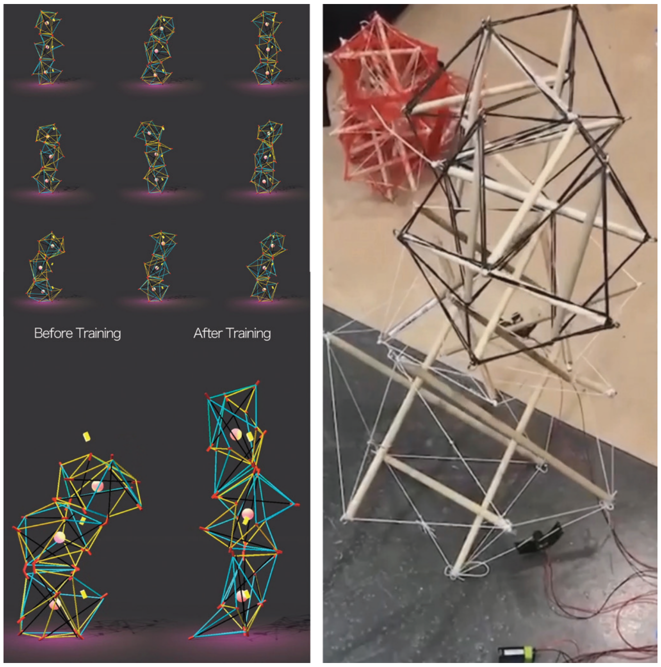
Reinforcement Learning was used to train multiple Deep Neural Networks to learn strategies for achieving objectives while being adaptive and robust to the stochasticity and variability of the environment. Parallel to our work, Reinforcement Learning for locomotion of tensegrity structures has been used before in projects like SUPERball (M. Zhang et al. 2017), however, our approach trains a plethora of additional policies to use in an architectural context.

We established an initial set of the following objectives related to autonomous architectural self-assembly:

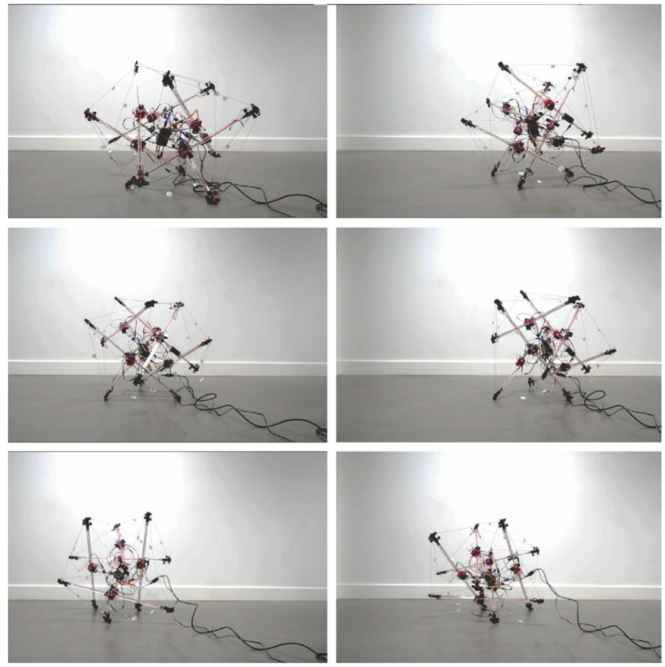
Locomotion: Enables units to autonomously move from one location to another over different terrains.
Multi-Agent Coordinated Locomotion: Multiple units coordinated locomotion to form a shape.
Self-Erection: The ability of a unit or multi-unit assembly to raise its centre of gravity up from the ground without pretensioning.
Localised Unit Connection: Enables multiple units in close proximity to connect to each other by aligning their nodes with tolerance.
Spatial Reconfiguration: Moving specific combinations of nodes towards a location for spatial adaptation while maintaining stability.
Multi-Agent Assembly Stability: Connecting of multiple units stacked vertically to balance each other.

For these objectives we designed a set of simple challenges to train a set of policies. All of the policies use the same Observations and Actions.

Observations: The states or observations consist of the tension of each of the 24 strings of the tensegrity structure, the velocity and position of each of the 12 nodes, and the



17 Multi-Agent Balance Reinforcement learning; Robot Multi-Unit Balance



18 Video Frames: Robot walking sequence translated from RL trained model.

three-dimensional vector representing the direction of the goal, relative to the centre of mass of the structure. The 99-dimensional vector represents the state of the agent. Actions: We chose to use continuous values to represent the actions of the agent. Each action directly controls the tension of each string of the structure. This forms a vector of 24 values controlled by the agent at each timestep.

The policy is a mapping (or a function) between the 99-dimensional state vector to the 24 valued action vector. This mapping π(a|s), where s is the state and an action as defined above, is modelled with a neural network of 3 hidden layers and 512 units each. The policy is trained using an algorithm called Proximal Policy Optimization (Schulman et al. 2017). The implementation we used was the one developed in Unity3D ML-Agents framework.

Below follows a list of the policies that we investigated:

Policy 1: Locomotion

This policy is responsible for varying the tension of the strings in order to get the tensegrity structure to move to a target. For this, we set the reward inversely proportional to the euclidean distance from the goal. To prevent overexcitement of the strings, we penalized the agent proportionally to the euclidean norm of the action vector. Higher values on the actuation of certain strings would increase the norm and decrease the reward, helping to avoid such behaviour. Finally, we added a time penalty of 0.05, to penalize for each timestep that the agent was not on the target to motivate the agents to reach the goal faster. The overall description of the reward is:

Reward_distance = 1/Distance
Penalty_energy = Norm(Action Vector)
Penalty_time = 0.05
Reward = Reward_distance - Penalty_energy
- Penalty_time

Policy 2: Balance to target

We designed the reward function to be inversely proportional to the target, while fixing nodes on the floor enabling units a localised "reaching" behaviour toward target nodes for connecting to nearby units.

Reward = Reward_distance - Penalty_energy

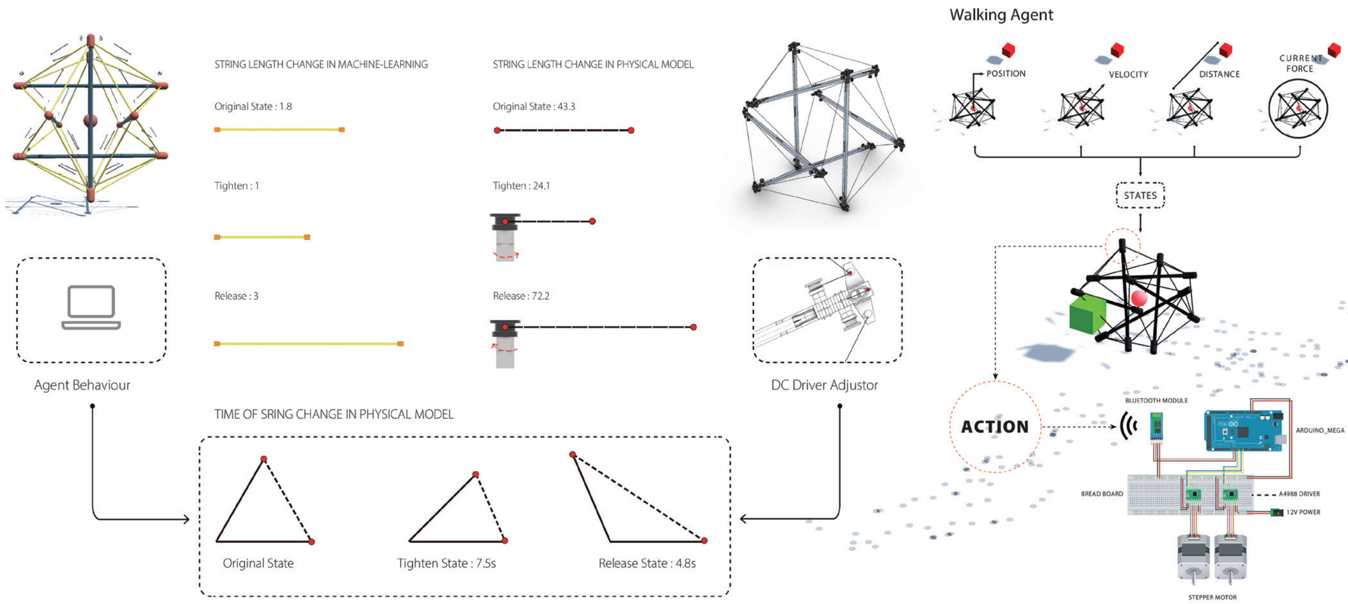
Policy 3: Multi-agent Balance

We connect multiple agents stacked on top of each other, forming a tower. Each of the agents is controlled by an independent policy trained in parallel. Each one uses the balance to target reward, while also using the observations and reward of the neighboring agents during training. Here weight is controlling the tradeoff between the agents reward and the global reward.

Reward = Reward_distance - Penalty_energy +
Sum_of_Neighbor_Agents_Rewards*weight

Policy 4: From ground to target

Here the agent learns how to change the tension of the strings, to stand up to a certain position starting from the



19 Diagrams: Interelationality between physical robots, simulated robots and reinforcement learning models

ground. The reward, in this case, is proportional to the height of the centre of gravity of the structure.

Reward = Center_of_mass_height - Penalty_energy

Policy 5: Multi-Agent ground to target

Similar to the single-agent ground-to-target, here the sum of the rewards of the neighboring agents are added to promote cooperation. The weight in this case controls the importance of the global reward.

Reward = Center_of_mass_height - Penalty_energy + Sum_of_Neighbor_Agents_Rewards*weight

RESULTS AND DISCUSSION

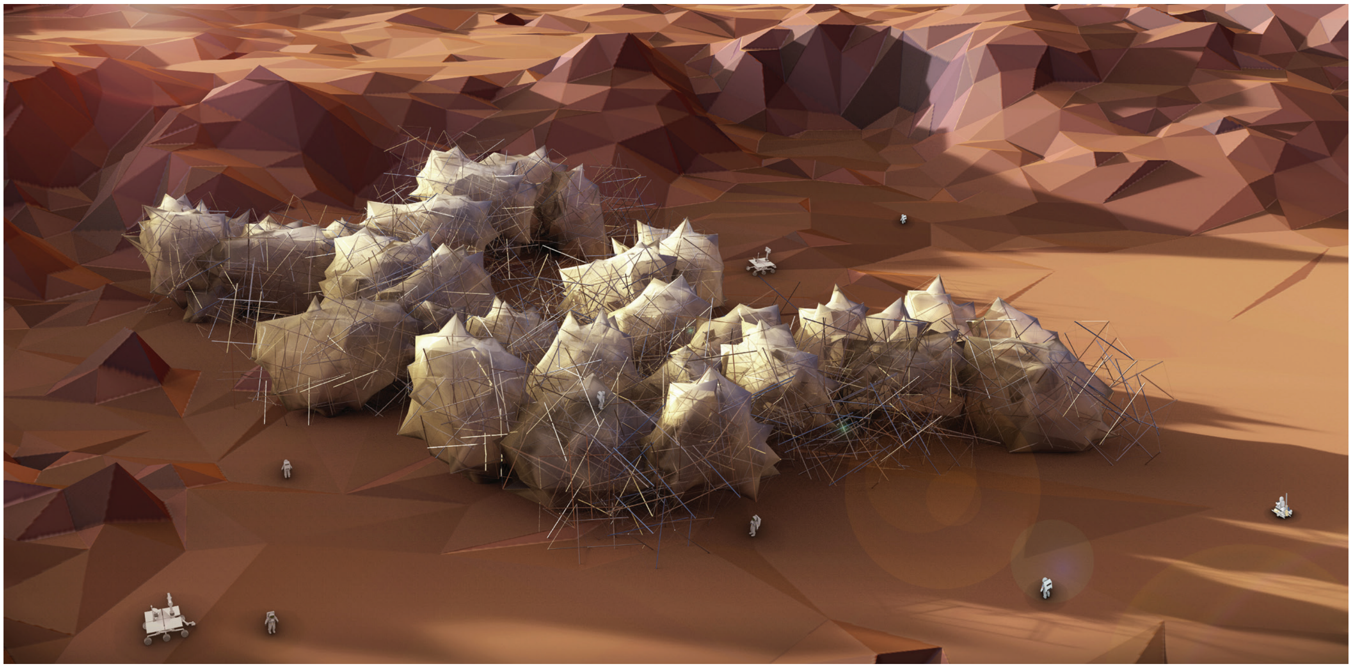
Autonomous Robotic Tensegrity has proven to be a robust case study for our computational methodology for autonomous architecture, successfully demonstrating the interrelationality of its three principles through a computational framework directly aligned with physically adaptive robotic tensegrity.

Significant contributions of this research rest in the introduction of deep reinforcement learning for architectural policies in multi-agent assembly systems. Although methods for automatic tensegrity topology generation have been developed, they typically result in small, simple units reaching a single state of equilibrium (Lee and Lee 2016; Xu, Wang, and Luo 2016; Nagase, Yamashita, and Kawabata 2016). We utilise reinforcement learning to find a wide range of balanced states of equilibrium potential for most tensegrity topologies without knowing pretensioned lengths of tensile elements. This offers an enormous potential for multi-objective form-finding in architecture, deriving efficient structures that can be assembled topologically on the ground and allowed to “self-erect” to many desired states. In addition to rapid self-erection, ART is capable of successful locomotion, localised connection for assembly, self-balancing equilibrium in multi-agent assemblies, and multi-agent coordinated reconfiguration.

After training, actuation locomotion sequences were recorded in the simulator and sent as instructions to the robot. Despite the simulation dynamics being inexact to the real dynamics, the robot prototype was able to move itself successfully using the simulated sequences.

Scaling up Reinforcement Learning to larger multi-agent assemblies is a challenging task since interaction between the agent and its environment is complex. Other agents are part of the environment with their behaviour being dependent on the behaviour of the rest of the agents as well. In the multiagent balancing task, the lowest level agent depends on the behaviour of the agents above it in the stack to succeed and cooperation is required.

Algorithms have been proposed to help with this problem, taking into consideration other agents’ actions (J. Foerster et al. 2018; Rashid et al. 2018; J. N. Foerster et al. 2016; Sukhbaatar, Szlam, and Fergus 2016) and also by sharing a common reward which all agents are trying to maximize.



20 Rendering: Speculative autonomous architecture proposal for Mars.

We found sharing rewards and observations between the agents was sufficient to demonstrate a multi-agent cooperation in the self-balancing task, however, scaling up to large numbers of agents proved challenging. Recent research in combining Graph Networks with Deep Reinforcement Learning for self-assembling morphologies (Pathak et al. 2019) has shown promising results in scaling up to larger multi-agent systems and we aim to explore these ideas.

ART has a master controller that chooses between policies in order to execute the desired task. For example, the master policy must first move the units to a common area, connect them, raise them up, and choose the balancing policies for stabilizing the structure executed in a sequence. To achieve true adaptability the sequential execution of adaptable sub-policies becomes itself an adaptable policy as well. Hierarchical Reinforcement Learning (Kulkarni et al. 2016) was not implemented here, but remains key in the ongoing research we are developing.

CONCLUSION:

ART shows potential for rapid autonomous deployment in undeveloped environments with limited equipment or resources and unforeseen on-site challenges. Autonomous architecture reappraises today's notion of construction in a linear building life cycle. Rather than further automating design, fabrication, construction as separate processes leading to static, inflexible forms, buildings are proposed as living systems that are scalable, reversible and self-adaptive with continuous life cycles, much like living ant colonies (Graham, 2017). By linking and embedding embodied and situated agency, facilitated variation, and intelligence in our design process we enable a more sustainable future adaptation of our buildings.

A shift toward autonomy is not simply an upgrade, but a disruptive technological and cultural change which suggests new questions be asked of how we design, construct and inhabit architecture. Key challenges with autonomous assembly include building performance, resilience, and engagement (Lu 2017). One challenge lies in these building components ability to form high performance enclosures rather than remain temporary structures or pavilions. Achieving high performance and resilience while maintaining facilitated variation enabled through effective modularity and weak-linkages of autonomous building parts is a key design challenge for truly scalable autonomous architectures.

The next step in our research is to develop multi-agent hierarchical reinforcement learning methods for training large assemblies with many units. This paradigm suggests a shift in the role of the architect to become the catalyst of intelligent processes through effectively designed reward functions. As Pask said, the role of the architect here, is not so much to design a building or city as to catalyze them; to act that they may evolve (Frazer 1995).

ACKNOWLEDGEMENTS

Living Architecture Lab, The Bartlett AD Research Cluster 3 (RC3)Studio Masters: Tyson Hosmer, David Reeves, Octavian Gheorghiu
Machine Learning Tutor: Panagiotis Tigas
Theory Tutor: Jordi Vivaldi Piera
Technical Tutor: Ziming He
team TARSS (2017-2018): Ziming He, Linlin Cao, Shi Ren, Ning Wang
Team Adaptive Tensegrity (2018-2019): Chi Siyu, Mengyi Lin, Jiachen Lei, Shiyu Kan
The Bartlett AD Director: Gilles Retsin
The Bartlett BPro Director: Professor Frédéric Migayrou
Main Softwares and Libraries used:
Unity3D, ML-Agents by Juliani, A. et al., SpatialSlur Library by David Reeves, Tensorflow Framework by Martin Abadi et al., McNeel Rhinoceros + Grasshopper, Karamba by Preisinger, Clemens, and Moritz Heimrath, Autodesk Maya, Autodesk 3ds Max, Adobe Photoshop, Adobe InDesign.

REFERENCES

Brand, Stewart. 1995. How Buildings Learn: What Happens after They're Built. Penguin.

Brooks, Rodney A. 1991. "New Approaches to Robotics." Science 253 (5025): 1227–32.

Foerster, Jakob, Richard Y Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. 2018. "Learning with Opponent-Learning Awareness." In Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, 122–30.

Foerster, Jakob N., Yannis M. Assael, Nando de Freitas, and Shimon Whiteson. 2016. "Learning to Communicate with Deep Multi-Agent Reinforcement Learning," 1–13. https://doi.org/10.7551/ecal.

Frazer, John. 1995. "An Evolutionary Architecture."

Gerhart, John, and Marc Kirschner. 2007. "The Theory of Facilitated Variation." Proceedings of the National Academy of Sciences 104 (suppl 1): 8582–89.

GlobalData. n.d. "Global Construction Outlook to 2022-Q4 2018 Update." https://www.orbisresearch.com/reports/index/global-construction-outlook-to-2022-q4-2018-update.

Graham, Jason M, Albert B Kao, Dylana A Wilhelm, and Simon Garnier. 2017. "Optimal Construction of Army Ant Living Bridges." Journal of Theoretical Biology 435: 184–98.

Ingber, Donald E. 1998. "The Architecture of Life." Scientific American 278 (1): 48–57.

Jenett, Ben, and Kenneth Cheung. 2017. "Bill-e: Robotic Platform for Locomotion and Manipulation of Lightweight Space Structures." In 25th AIAA/AHS Adaptive Structures Conference, 1876.

Kober, Jens, J Andrew Bagnell, and Jan Peters. 2013. "Reinforcement Learning in Robotics: A Survey." The International Journal of Robotics Research 32 (11): 1238–74.

Kulkarni, Tejas D, Karthik Narasimhan, Ardavan Saeedi, and Josh Tenenbaum. 2016. "Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation." In Advances in Neural Information Processing
Systems, 3675–83. 6

Lee, Seunghye, and Jaehong Lee. 2016. "A Novel Method for Topology Design of Tensegrity Structures." Composite Structures 152: 11–19.

Levin, Stephen M. 2006. "Tensegrity: The New Biomechanics." Textbook of Musculoskeletal Medicine 9.

Levine, Sergey, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. 2016. "End-to-End Training of Deep Visuomotor Policies." The Journal of Machine Learning Research 17 (1): 1334–73.

Li, Yuxi. 2017. "Deep Reinforcement Learning: An Overview." ArXiv Preprint ArXiv:1701.07274.

Lu, Andong. 2017. "Autonomous Assembly as the Fourth Approach to Generic Construction." Architectural Design 87 (4): 128–33.

Mao, Hongzi, Mohammad Alizadeh, Ishai Menache, and Srikanth Kandula. 2016. "Resource Management with Deep Reinforcement Learning." In Proceedings of the 15th ACM Workshop on Hot Topics in Networks, 50–56.

Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. "Playing Atari with Deep Reinforcement Learning." ArXiv Preprint ArXiv:1312.5602.

Nagase, K, T Yamashita, and N Kawabata. 2016. "On a Connectivity Matrix Formula for Tensegrity Prism Plates."

Mechanics Research Communications 77: 29–43.

Ngwepe, Lusca, and Clinton Aigbavboa. 2015. “A Theoretical Review of Building Life Cycle Stages and Their Related Environmental Impacts.”

Parter, Merav, Nadav Kashtan, and Uri Alon. 2008. “Facilitated Variation: How Evolution Learns from Past Environments to Generalize to New Environments.” PLoS Computational Biology 4 (11): e1000206.

Pathak, Deepak, Chris Lu, Trevor Darrell, Phillip Isola, and Alexei A Efros. 2019. “Learning to Control Self-Assembling Morphologies: A Study of Generalization via Modularity.” ArXiv Preprint ArXiv:1902.05546.

Petersen, Kirstin Hagelskjaer, Radhika Nagpal, and Justin K Werfel. 2011. “Termes: An Autonomous Robotic System for Three-Dimensional Collective Construction.”

Petersen, Kirstin, and Radhika Nagpal. 2017. “Complex Design by Simple Robots: A Collective Embodied Intelligence Approach to Construction.” Architectural Design 87 (4): 44–49.

Popescu, George A, Tushar Mahale, and Neil Gershenfeld. 2006. “Digital Materials for Digital Printing.” In NIP & Digital Fabrication Conference, 2006:58–61.

Rashid, Tabish, Mikayel Samvelyan, Christian Schroeder De Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. 2018. “QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning.” ArXiv Preprint ArXiv:1803.11485.

Retsin, Gilles. 2019. “Bits and Pieces: Digital Assemblies: From Craft to Automation.” Architectural Design 89 (2): 38–45.

Retsin, Gilles, and Manuel Jimenez Garcia. 2016. “Discrete Computational Methods for Robotic Additive Manufacturing: Combinatorial Toolpaths.”

Reynolds, C.W. 1987. “Flocks, Herds and Schools: A Distributed Behavioral Model.” In ACM SIGGRAPH Computer Graphics, 21:25–34. ACM. https://doi.org/10.1145/37402.37406.

Rubenstein, Michael, Christian Ahler, and Radhika Nagpal. 2012. “Kilobot: A Low Cost Scalable Robot System for Collective Behaviors.” In 2012 IEEE International Conference on Robotics and Automation, 3293–98.

Russell, Stuart J, and Peter Norvig. 2016. Artificial Intelligence: A Modern Approach. Malaysia; Pearson Education Limited,.

Sukhbaatar, Sainbayar, Arthur Szlam, and Rob Fergus. 2016. “Learning Multiagent Communication with Backpropagation,” no. Nips. http://arxiv.org/abs/1605.07736.

Sutton, Richard S, Andrew G Barto, and others. 1998. Introduction to Reinforcement Learning. Vol. 135. MIT press Cambridge.

Tibbits, Skylar. 2012. “Design to Self-Assembly.” Architectural Design 82 (2): 68–73.

Tibbits, Skylar. 2017. “From Automated to Autonomous Assembly.” Architectural Design 87 (4): 6–15. 6

Xu, Xian, Yafeng Wang, and Yaozhi Luo. 2016. “General Approach for Topology-Finding of Tensegrity Structures.” Journal of Structural Engineering 142 (10): 4016061.

Zhang, Li-Yuan, Song-Xue Li, Shi-Xin Zhu, Bo-Yang Zhang, and Guang-Kui Xu. 2018. “Automatically Assembled Large-Scale Tensegrities by Truncated Regular Polyhedral and Prismatic Elementary Cells.” Composite Structures 184: 30–40.

Zhang, Marvin, Xinyang Geng, Jonathan Bruce, Ken Caluwaerts, Massimo Vespignani, Vytas SunSpiral, Pieter Abbeel, and Sergey Levine. 2017. “Deep Reinforcement Learning for Tensegrity Robot Locomotion.” In 2017 IEEE International Conference on Robotics and Automation (ICRA), 634–41.

IMAGE CREDITS

Tensegrity Icosahedron, Bob Burkhardt, Line drawing made by Bob Burkhardt using Computervision CADDS program, 1994.

Figures 1,6,8,9,10,12,13,14,16,17,18,19,20. Living Architecture Lab, The Bartlett AD-RC3, 2017-18 Team: TARSS

Figures 2,3,4,5,7,15. Living Architecture Lab, The Bartlett AD-RC3, 2018-19, Team: Adaptive Tensegrity

Tyson Hosmer is a Course Master of Research Cluster 3, directing the Living Architecture Lab at UCL Bartlett BPro in London. He is an Associate at Zaha Hadid Architects where he leads the ABPS group. His research at the office focuses on the application of machine learning with agent based systems. He has over twelve years of experience working in international architecture offices including Axi:Ome, Asymptote Architecture, Kokkugia, and Cecil Balmond Studio where he was the Research Director for several years. Before joining the Bartlett, he taught for six years with the AA Design Research Lab. Tyson is interested in embedding localized artificial intelligence in nonlinear organizational systems applied in a design process.

Panagiotis Tigas is a PhD student and researcher at the University of Oxford, focusing on Artificial Intelligence, Generative Design and Robotics. In the past, he has worked for Microsoft, Autodesk Research and also several startups.