Scalable Robotic Voxel Construction

by

Miana M. Smith

S.M., Massachusetts Institute of Technology (2023)
S.B. Massachusetts Institute of Technology (2021)

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2026

©2026 Miana Smith. All rights reserved
The author hereby grants to MIT a nonexclusive, worldwide, irrevocable, royalty-free license to exercise any and all rights under copyright, including to reproduce, preserve, distribute and publicly display copies of the thesis, or release the thesis under an open-access license.

Authored by:
Miana Smith
Program in Media Arts and Sciences
August 21, 2026

Certified by:
Neil Gershenfeld
Director
MIT Center for Bits and Atoms
Thesis Supervisor

Accepted by:
Joseph Paradiso
Academic Head
Program in Media Arts and Sciences

Scalable Robotic Voxel Construction
by
Miana M. Smith

Submitted to the Program in Media Arts and Sciences, School of Architecture and Planning, on August 21, 2026, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Media Arts and Sciences

Abstract

This thesis advances distributed robotic assembly of structural digital materials as a route to automated, in-situ building construction with lowered embodied carbon, cost, and build times. Automated fabrication could substantially improve the sustainability and efficiency of construction, but current methods fall short at the building scale. Continuous processes, such as large-scale 3-D printing, are limited by accessible build volume, a narrow set of printable materials, and an inability to correct errors, which can accumulate into irreparable failures. The resulting structures then offer little or no option for modification, repair, or end-of-life reuse. Assembly-based processes instead demand high absolute mechanical precision, pushing the work toward large, complex, and expensive machines, or toward off-site prefabrication and transport. In both cases, cost, speed, structural performance, design freedom, and material sustainability are traded off against one another. Distributed robotic systems for digital material systems offer an alternative approach, where multiple small robots cooperatively construct large structures by traversing and building directly on the workpiece using an error-correcting material system. However, prior systems have struggled to achieve sufficient load-bearing capacity, assembly throughput, and geometric flexibility to scale to real-world building scenarios.

This work develops new structural discrete lattices where the full logic of construction, spanning material manufacturing, assembly, structural performance, and other functional requirements, is embedded directly into the material system itself such that it facilitates the cost-effective assembly of sustainable and load-bearing structures using simple robots. First, I introduce electronics-integrated voxels that carry power, signal, and compute, serving as the basis for direct-write electromechanical structures and for modular robots that assemble further robots, towards recursive, self-scaling swarms. Next, I introduce hierarchical interlocking voxel architectures and a new modular assembler that together improve assembly throughput, mechanical performance, and material efficiency while remaining material-agnostic and mass-manufacturable from flat feedstock, including from renewable materials. Finally, I evaluate the system at the architectural scale through a comparative study against existing construction methods, including framing and concrete-based approaches, showing a 76–82% reduction in embodied carbon relative to 3-D printed concrete, and through the construction of load-bearing structures all built from reusable parts and renewable materials, including a 2 m x 2 m x 2 m prototype house. Together, these results demonstrate a path toward sustainable and functional architectural scale buildings assembled by self-scaling, distributed robotic systems.

Thesis Supervisor: Neil Gershenfeld
Title: Director, MIT Center for Bits and Atoms

Scalable Robotic Voxel Construction

by

Miana M. Smith

The following people served as readers for this thesis:

Caitlin Mueller
Associate Professor
Architecture and Civil and Environmental Engineering
Massachusetts Institute of Technology

Maria Yablonina
Assistant Professor
John H. Daniels Faculty of Architecture, Landscape, and Design
University of Toronto

Contents

List of Figures

List of Tables

Acknowledgments

I will start by thanking my advisor, Neil Gershenfeld, who has created in the CBA a strange place for strange people to do strange research (positive!). I was introduced to the power of digital materials and digital fabrication as an undergrad in your class How to Make (Almost) Anything and that's just been lodged in my brain ever since as possibly the solution to all of our environmental and economic challenges (slight exaggeration).

I'd next like to thank my committee. Caitlin Mueller, thank you for bringing the real world structural perspective to this work. You've been a part of my committee since my Master's and your perspective has really shaped this work in impactful ways. Maria Yablonina, thank you for representing the distributed robots perspective. I've really appreciated all of the conversations we've had and the feedback you've provided. I'd also like to acknowledge committee members from earlier in this journey, Russ Tedrake during the master's, and Kirstin Petersen during quals, as your perspectives have also shaped this work a good deal.

Research is a group effort that extends off the existing body of work developed over... a really, really long time. I want to acknowledge and thank my collaborators who helped with developing the work in this thesis: Amira Abdel-Rahman, Paul Richard, Alessio Zazo, Jack Forman, Alfonso Parra Rubio, Alexander Htet Kyaw, Chris Cameron, Vlasta Kubusova, Kristof Pucedl, Sergio Mutis, the UROPs: Sophia Wang, Andy Dequin, and Ryan Hoffman, and my direct-ish research lineage: Ben Jenett, Christine Gregg, and Kenny Cheung. This isn't possible without all of your work. I'd like to additionally call out all of the mentorship and support that Amira has provided over the years— thank you, you've been the shadow 4th committee member over my entire time at CBA.

In addition to my immediate collaborators, the direct and extended CBA ecosystem has provided a huge amount of support throughout the years. Alan, Eyal, Anthony, Eric, Sara, another Sara, Jiaming, Marcello, Camron, Alex, Dimitar, Quentin, Jake, Zach, Natalie, Patricia, Jiri, Kat— thank you all.

And to the people who actually make CBA run— Dan, Kara, Candace, Marissa— thanks for keeping it going. And to Sherry and the Fab Foundation folks— thanks for all those opportunities.

To the various sponsors of this research— the Bhutanese team at DHI and the Superfablab, the Toyota teams led by Kohshi Katoh and Tsuyoshi Izuhara, the Autodesk Innovation Center team— thank you for your support.

To my earlier research mentors: Larry and Rachel in undergrad, and Pat, Scott, and Joe before that. And earliest of all, to Carver, thank you for sparking my love of science.

Dave— thank you for the answers to hard hitting technical questions, your cool desk space, and for your friendship, it's been transformative.

Serena, Jesse, Yiran, Claire 'n Mau, Irene— thank you guys for the adventures (and bonus thanks to Serena for being my og first reader).

To Anna Chi and Douglas Smith (my parents), and the whole Smith-Chi family, thank you all for all of your love, support, food, and strange art through the years. It means a lot.

And of course: thank you to my partner in life (and wife!) Milka Piszczek. None of this could happen without you. I love you.

Chapter 1

Introduction

Modern computing (and all that this has enabled) is predicated on the concept of encoding analog signals with digital representations. In "Communication in the Presence of Noise" (1949), Claude Shannon showed how the conversion of an analog system into a digital one enables communication systems to overcome the limitations of noise [158]. While an analog signal would be directly corrupted by noise distortions, converting it to a digital signal enables reliable communication through inherent error correction. So long as the noise in a given channel stays below a certain threshold, the original message can be perfectly reconstructed. This shift from analog to digital enabled reliable, accurate, and scalable communications in spite of unreliable and imperfect hardware.

As computing systems matured, they eventually converged with the evolution of machine automation for physical manufacturing processes. The numerical-control (NC) machines developed in the mid 1900s, and the computer numerical-control (CNC) machines that followed, used digital instructions to guide the precise manipulation of material into accurate and repeatable geometries. Current day digital fabrication systems largely follow this same paradigm, in which designs are encoded as digital information and executed using computer-controlled machines to shape a fundamentally analog material through additive or subtractive processes.

Unlike traditional fabrication processes, digital fabrication is typically better able to decouple design complexity from production cost. This can enable the production of more optimal, custom, or otherwise high-performance geometries. This is becoming increasingly common for parts on the order of 10s of centimeters in scale: additive metal printing in combination with 5-axis machining, for example, offers an efficient route toward complex, high-performance parts currently used for aerospace applications [99]. Similarly, stereolithography printing is widespread in the dental industry, where the ability to produce highly detailed and bespoke models offers a significant advancement over traditional workflows [25]. This ability to produce complex parts with minimal tooling set up additionally enables faster design iteration, which can lead to better performance.

Although many digital fabrication systems are already quite mature and prolific at the desktop scale or below, we see much less application at physically large scales. This largely stems from the fundamental challenge of manipulating analog materials accurately at large scales, while maintaining practical speed, cost, and geometric adaptability. So, while digital fabrication has been presented as a potential route towards increased efficiency, innovation, and productivity in many areas [99], its ability to achieve this has been quite limited for large scale systems.

This ability to produce more complex geometries is significant toward improving how we build big things, especially in construction. The construction industry drives 38% of global greenhouse gas emissions in energy [22] and construction demand is projected to continue increasing [54]. By enabling more material-efficient designs with more diverse materials, digital fabrication could enable more sustainable building practices [66]. For example, the largest global use of steel is in construction, and a plurality of that use goes toward reinforcement structures for concrete (rebar). However, most rebar structures use substantially more material than necessary to accommodate manual building practices, leading to more unified bar sizes and square grids, as opposed to more structurally optimal configurations [3]. If design complexity were decoupled from cost and time, as it would be in a theoretical digital fabrication system, it would likely enable more material efficient solutions, leading to both sustainability and cost improvements. Automated on-demand construction of more custom geometries would also improve the overall efficiency of construction, which is currently low, and has remained relatively stagnant as compared to other industries [175] [66].

However, current methods for digital fabrication at the meter-scale and beyond fall short at realizing this vision of adaptable, on-demand, and sustainable building. The competing demands of precision, speed, and structural performance tend to result in large, expensive, and often under-performing machine systems. Small errors can accumulate over large distances to become very large errors and in continuous deposition processes, such as 3D concrete printing, this can result in irreparable failure [41]. Assembly based approaches, in contrast, such as timber assembly in [172] or [135], can be repaired, though in the absence of an error-correcting material system, can require even more precise (and expensive) robotic systems for assembly.

Purpose-built robot-material systems offer an alternate approach that can sidestep many of the challenges that industrial machines and robotics face when adapted for large scale fabrication systems. Because building material and robot are designed specifically to work with each other, the requirements on the actuation, sensing, and computation systems of the robotics are substantially lowered [93]. This lowered complexity (which in many ways is coupled with lowered cost) opens parallelization as a route toward scaling [17]. This can enable a set of relatively simple robots to build structures much bigger than themselves, at rates competitive with static machine systems [164].

This is enabled by the use of digital material systems (whether or not all of these works would classify their material systems this way). The shift to digital logic enabled the creation of much more complex computational systems because it introduced a fundamental building

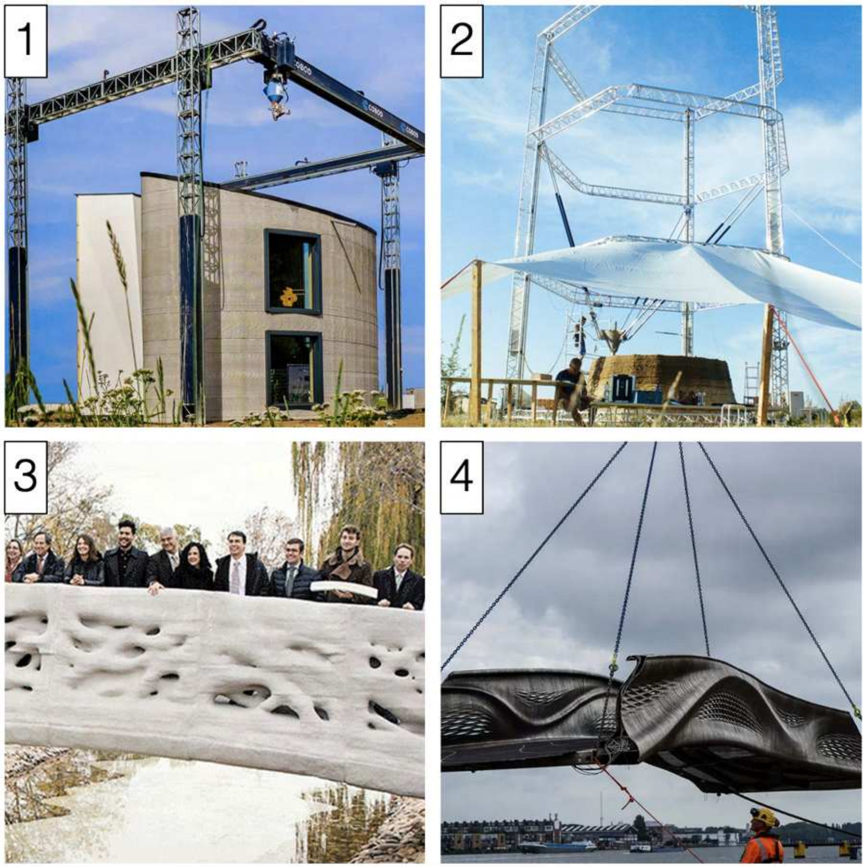

Figure 1-1: Examples of large-scale continuous digital fabrication systems. 1) The COBOD BOD2 concrete printer has been used to produce the walls of actual houses; however, their standard practice is to use the printed elements as a stay-in-place formwork that traditional concrete is cast into[34] Image from [49] 2) The WASP big Delta printer, which explores mud printing as a more sustainable alternative to concrete printing. Image from [12] 3) IAAC concrete printed pedestrian bridge. This bridge was pre-printed in sections and delivered as a large module to the site, which is then tensioned together. Image from [84]. 4) Welded steel pedestrian bridge by MX3D, printed off-site and delivered as a final component. The difficulties of the manufacturing process resulted in substantial material overuse. Image from [10]

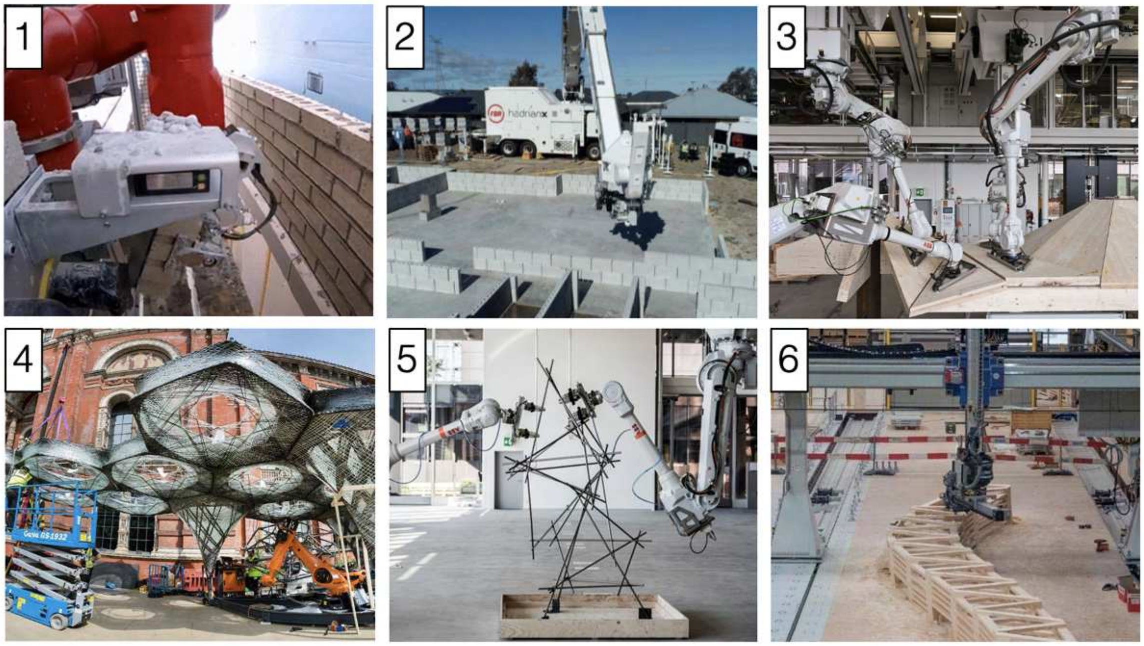

Figure 1-2: Examples of large scale assembly based systems. 1) SAM bricklayer, which partially automated red clay brick placement. Image from [11]. 2) The HadrianX bricklayer, which places mortar-less CMUs from a telescoping multi-DoF arm. Image from [13]. 3) Collaborative wood panel installation by multiple industrial arms from [154] 4) Elytra filament pavilion, featuring wound fibers by an industrial arm [146]. 5) Collaborative construction of metal structures from [136] 6) Gantry-based assembly of wood truss elements from [26]

block which could reject external noise. Similarly, a digital material system is able to reject noise (an imprecise assembler) and prevent error accumulation, enabling the assembly of larger and more complex structures [89]. This inherent error correction in the material system has been achieved in a myriad of ways ranging from passive mechanical features [163] to electronic actuation [106]. Figure 1-3 plots selected distributed robotic assembly systems from the literature along an axis indicating degree of robot-material integration and along an axis indicating material complexity. In general, error correction features will introduce some amount of additional complexity into the material system. This tradeoff of complexity between the material system and the robots often constrains the ability of these systems to build either large [78], load-bearing [142], or geometrically complex [121] structures, which have so far limited the deployment of these kinds of material-robot systems to practical real-world applications. This is because the overall manufacturing of the system becomes prohibitively complex to enable assemblies at the meter scale and above, with practical load bearing capacity and sufficient design freedom.

Prior large scale fabrication systems ranging from static gantry systems as in Figure 1-1, to assembly approaches that re-purpose industrial robotics as in Figure 1-2, to distributed robots that build much bigger structures than themselves such as those in Figure 1-3 all

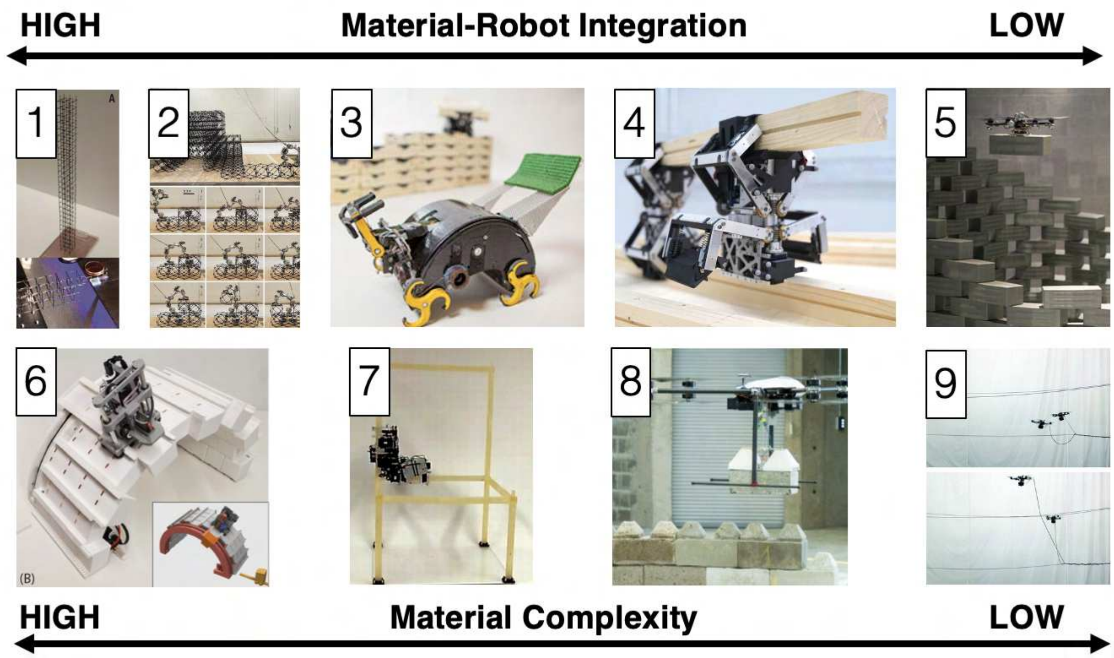

Figure 1-3: Considering the level of material-robot integration as well as material complexity for an example set of distributed robotic assembly systems. While higher degrees of material-robot integration often enable simpler robotic systems, these often push complexity onto the material system instead, making it harder to scale to higher unit counts, and therefore larger structures. Images from: 1) carbon fiber truss assembling microrobots [78]. 2) SOLL-E voxel assembler from [71]. 3) TERMES magnetic block stacking whegged robots from [142]. 4) Rotational clamp-based wood beam assembler that use the building material as part of the robotic kinematic chain, from [112]. 5) Foam brick stacking collective quadcopter assembly from [28]. 6) A brick based vault building robotic system from [121]. 7) An reconfiguring truss-based assembler robot from [131]. 8) Full-sized concrete brick stacking by a UAV from [69]. 9) Load-bearing filament winding by collaborative quadrotor drones from [125].

contend with a set of often competing design parameters, which can give rise to set of guiding design principles. Each of these tensions is resolved by shifting functionality off the robot and onto the material system, such that it can embody the needed structural, performance, or assembly degrees of freedom:

- **Machine size vs. build volume:** Unlike most existing approaches, our goal is to de-couple machine or robot size from the overall build volume of the system. In this way, our system can be more broadly applicable to a wider array of applications and environments.
- **Robotic complexity vs. reliability**: Often, the reliability of a robotic assembly system is coupled to its complexity, making it costly and potentially infeasible to

parallelize or otherwise scale it to more practical applications. So, we aim to create a simple, low-cost robot.

- **Material complexity vs. mechanical performance**: In existing distributed robotic assembly approaches, systems with the greatest load-bearing capacity typically rely on highly complex material systems. This limits the overall system performance, so our goal is to lower the material complexity without sacrificing mechanical performance.

- **Material efficiency, mass production, and sustainability:** While these goals are not necessarily in tension with each other, as the prior ones often are, they also are not quite well-coupled with each other. Guaranteeing material efficient designs often makes mass-production difficult, and while material-efficiency can result in sustainability benefits, this is also highly contingent on the production method, especially for complex geometries.

To address these challenges, this thesis explores digital fabrication, as in: fabrication that utilizes digital material systems, as a strategy for addressing these scaling challenges. Much like the shift from analog to digital computation was enabled by inherent error-correction mechanisms that manage unreliability of physical systems, this work considers the mechanical counterparts of such error-correcting systems for fabrication. Discretizing a structure into inherently error-correcting elements can create fabrication processes that are more robust to noise and uncertainty during assembly. However, this considers only the assembly logic of the structural element. In practice, each element in a structure embodies and performs multiple functions. Beyond embodying the assembly logic, it is also necessary to consider the element's structural logic, how it is manufactured, and any other functional or design requirements. By creating digital elements that embody each of these functionalities, we can realize systems that are possibly fundamentally more suited to reliable, scalable, large-format digital fabrication.

The central argument of this thesis is that the logic of construction should be embedded in the building block itself, rather than in the machine that places it. That is, how a structure performs, how it is made, and how it is assembled, can be encoded into simple blocks to realize scaleable robotic assembly systems for large scale structure. This work advances large-scale digital fabrication by developing scalable distributed robotic assembly systems based on discrete lattice building elements, or voxels, with integrated structural, electronic, and/or robotic functionality. Through new electronics-integrated hardware, hierarchical interlocking architectures, and demonstrations of robotic self-assembly and adaptable robot geometries, this work demonstrates how digital material systems can potentially better address the precision, scalability, and reliability limits of analog large-scale fabrication. The material-robot system is then extended to architectural-scale evaluations to help validate it as a practical and sustainable path toward distributed robotic construction.

The specific contributions and their appearance in the thesis are as follows. Each of these explores a different design degree of freedom integrated into the material system.

- **Scalable voxel-based robotic material systems:** Development of modular, lattice-based building blocks that integrate structural, electrical, and/or robotic functionality. (Throughout)
- **Electronics-integrated voxel hardware:** New hardware enabling direct-write electromechanical fabrication and supporting modular robotic assembly. (Chapters 2 and 3)
- **Robotic self-assembly demonstrations:** Demonstrations of robotic assembly of other robot geometries, toward self-scaling or adaptive distributed robotic assembly systems. (Chapter 3)
- **Hierarchical voxel architectures:** Introduction of voxel designs that increase assembly speed, reduce cost, and improve scalability for large structures through hierarchical grouping and interlocking of lattice elements, while achieving efficient material usage and maintaining compatibility with a range of manufacturing options and materials. (Chapters 4, 5, and 6)
- **Assembler Robots:** Reliable, low-cost relative robots for voxel assembly with order-of-magnitude increase in volumetric assembly rate compared to the prior art. (Chapters 4 and 6)
- **Architectural-scale evaluation:** A comparative study against existing automated and manual construction methods, and demonstration of the autonomous construction of architecture-scale load-bearing structures. (Chapters 5 and 6)
- **Overall contribution:** A route toward distributed, error-correcting, self-scaling robotic construction systems capable of fabricating large, functional, load-bearing structures with reduced material use and minimal fixed infrastructure via voxel systems that embed their own structural, manufacturing, and assembly logic in their design.

Chapter 2

Functional Voxels for Prototyping Electronic Structures

The work and text in this chapter appear in part in the publication "Voxel Invention Kit: Reconfigurable Building Blocks for Prototyping Interactive Electronic Structures" which was written with my co-first author Jack Forman, as well as Amira Abdel-Rahman, and Sophia Wang. This was published in CHI Conference on Human Factors in Computing Systems (CHI '25).

2.1 Abstract

Prototyping large, electronically integrated structures is challenging and often results in unwieldy wiring, weak mechanical properties, expensive iterations, or limited reusability. While many electronics prototyping kits exist for small-scale objects, relatively few methods exist to freely iterate large and sturdy structures with integrated electronics. To address this gap, we present the Voxel Invention Kit (VIK), which uses reconfigurable blocks that assemble into high-stiffness, lightweight structures with integrated electronics. We do this by creating cubic blocks composed of PCBs that carry electrical routing and components and can be (re)configured with simple tools into a variety of structures. To ensure structural stability without expertise, we created a tool to configure structures and simulate applied loads, which we validated with mechanical testing data. Using VIK, we produced devices reconfigured from a shared set of voxels: multiple iterations of a customizable AV lounge seat, a dance floor game, and a force-sensing bridge.

2.2 Introduction

Functional interactive devices can be difficult to prototype, especially at sizes larger than the desktop scale. To help illustrate the friction in current standard digital fabrication

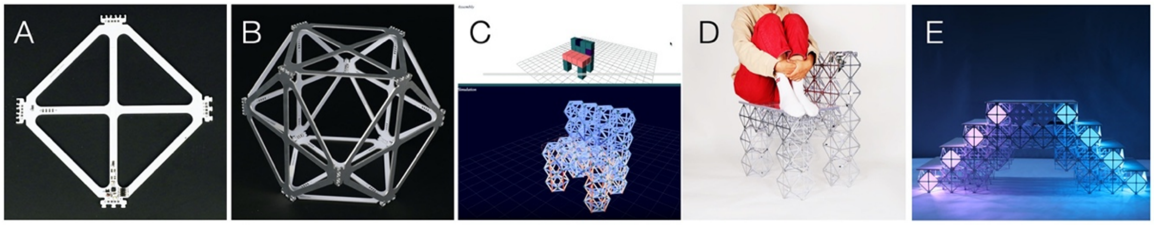

Figure 2-1: An overview of VIK. A) a single lattice face produced in a PCB house; B) an assembled functional voxel; C) a screenshot of the design and simulation tool; D) an assembled responsive chair; E) a pressure sensing bridge assembled from VIK.

workflows, we consider a hypothetical scenario: Victor is a designer who wants to prototype an interactive chair as part of their course final. Though Victor is hypothetical, this story is informed by the many years we have spent teaching a digital fabrication course, and the common struggles we have observed students encounter while trying to deploy functional objects. Victor begins by using CAD to design the mechanical elements of the chair, intending to eventually CNC cut the chair from wood. Victor first makes a prototype from laser-cut cardboard at their university maker space. After cutting, Victor realizes that the slot-fits are misaligned and has to re-design and re-cut, wasting the previous material and shop-time. Victor then prototypes the electrical system by placing different modules on the chair and connecting them via a mix of jumper cables and copper tape, which have to be soldered as the connections aren't robust enough. Victor then moves on to a final version using $\frac{1}{2}$" plywood, which is thicker than necessary to guarantee load-bearing capacity. They then have to adjust the design, re-cut, re-assemble, and re-integrate the electronics. While carrying the chair to class, a copper tape trace is torn on a door handle, and Victor has to last-minute debug and fix the chair. After the demo, some students have suggestions as to how the chair design and electronics could be improved, but Victor won't follow up on these suggestions as they would require substantially reworking the chair. In the end, Victor moves the chair to their office, before eventually throwing it out once the electronics stop reliably working.

Instead of this laborious process, an ideal combined mechanical and electrical prototyping system would afford cheap, quick iterations with simple tools, robust mechanical and electrical properties, and various I/O capabilities. However, the development of such a system comes with three main challenges: 1) the prototyping of large and robust structures is too time-consuming and/or expensive for rapid iteration; 2) approaches for embedding electronics into 3D forms are difficult or impossible to reconfigure, while the grafting of electronics onto 3D forms is typically fragile and inelegant; and 3) iterative prototyping of electromechanical systems requires continuous access to digital fabrication tools.

While prior work has addressed some issues associated with porting electronics to 3D forms, most of this work is targeted at a relatively small scale and wouldn't help our hypothetical designer. For example, at the desk-top scale, [185] uses curved breadboards for more accurate looks-like prototyping of electronic devices, [171] uses commercial metal 3D printing services to create functional components with embedded sensing elements, and [98] creates a workflow for conforming circuits to curved surfaces. And, while prior works have developed tools for large-scale mechanical prototyping systems, such as large-scale dynamic constructions in [104], or a handheld extruder for 3D-printing prototype furniture-scale objects [21], these leave electronics integration as an afterthought.

We present VIK, the Voxel Invention Kit, as a potential solution. VIK uses aluminum PCBs as 3D lattice building blocks, or voxels, that assemble into structures (see Figure 2-1 for an overview). The system's basic unit has high mechanical performance with embedded electronics and uses a cheap, fast, and commercially available manufacturing process to increase accessibility to the system.

Our main contributions are:

- The creation of an integrated prototyping platform for creating large and robust yet lightweight electromechanical structures that can be built and reconfigured cheaply, quickly, without waste, and using only a soldering iron and pliers.

- The development of press-fit functional voxels composed of high-stiffness and high-strength lattice units with integrated sensing, response, and processing abilities for assembling reconfigurable electromechanical structures, as well as electrical and physical accessories to customize IO capabilities and structural shape.

- A design tool to enable end-to-end rapid prototyping with VIK: users can use the tool to design structures and simulate the structure's response to mechanical loads to inform their design.

- The demonstration of potential interactive use cases with the voxels, specifically, we demonstrate the interactive prototyping process through several iterations of an IO-enabled chair, a dance floor gamepad, and a force-monitoring bridge, which are (re) assembled from a shared set of voxels.

2.3 Related Works

Few works in HCI focus on prototyping large and sturdy structures with embedded electronics. So, to further inform the development of our system, we draw from three topics in HCI that have received more extensive investigation: modular electrical prototyping kits, embedding electronics into 3D forms, and fabricating large, sturdy structures. In the following section, we investigate each of these sub-areas to identify their trends, successful approaches, and overarching limitations. A comparison of some key parameters is shown in Figure 2-2. These are then synthesized together to construct an intersectional set of principles used to develop VIK.

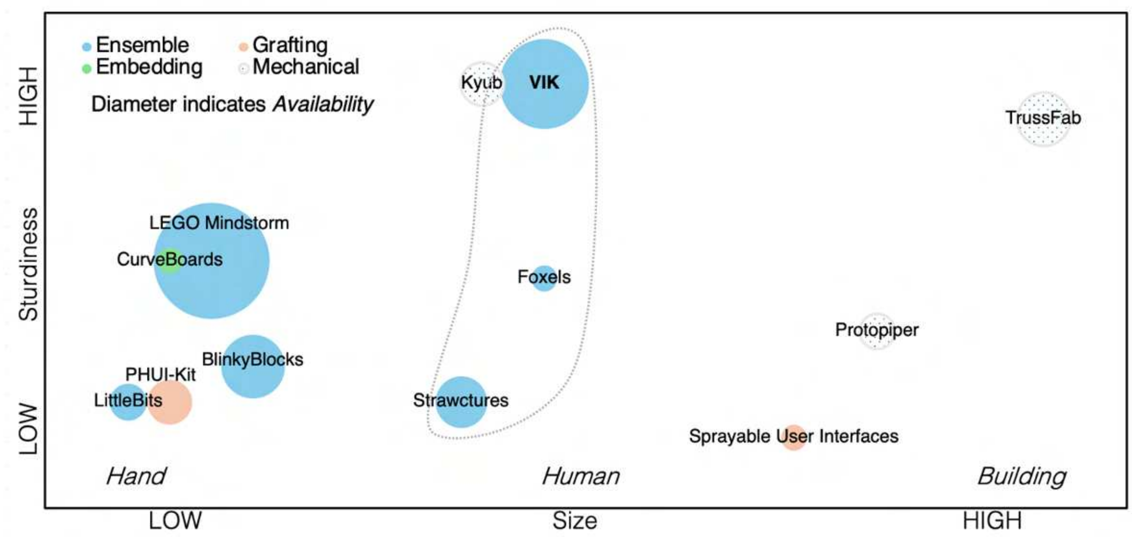

Figure 2-2: Comparison of selected electrical and mechanical prototyping kits in terms of sturdiness, size, and availability, where availability is a heuristic for how easy it is to acquire or create a practical quantity of parts for each kit.

2.3.1 Electrical Prototyping Kits

When designing modular electronic kits, there are critical design decisions and tradeoffs that affect the audience and agency a kit provides. We intend VIK to be a low-threshold high-ceiling prototyping platform for electromechanical structures that is targeted toward makers and practitioners. To inform the design of our system, we look to review papers that compare many types of prototyping kits to extract meta-themes. We prioritized reviews that discuss the prototyping kits for practitioners and makers, instead of those used for educational purposes. For educational kits, the focus largely lies on the learning that occurs through making, whereas maker-oriented kits emphasize what can be made with that system.

In a survey paper of electrical prototyping systems, Lambrichts et. al [108] identify several key takeaways. First, the top feature that users look for in these kits is that prototypes are easy to iterate— after that, users prioritize easy debugging, durability, reusability, and affordability. With this the authors postulate that “it is more convenient to construct a prototype with a toolkit that (1) does not require connecting individual wires, (2) can be programmed by physically interconnecting blocks, (3) does not require additional tools for connecting modules, such as adhesives or stitches, and (4) can be connected in a bus topology” [108]. Additionally, it is noted that people who employ electrical prototyping kits are generally resistant to adopting new platforms— they are happy with what they use, fearing that a new system will be inconvenient to learn, offer few feature improvements, and could be deprecated in the future. Thus, even if a new system is easier to use, that does not mean that people will actually use it. They also note that for users already familiar with electrical prototyping kits, the electrical and programming expertise required by existing systems is rarely a major barrier and suggest focusing on other features to improve.

2.3.2 Fabricating 3D Forms with Integrated Electronics

Since VIK is intended as a combined mechanical and electrical prototyping system, we investigated the different ways that people integrate electronics into physical devices. From this, we identified three main themes: grafting, embedding, and ensemble configuration. In this section, we sift through the considerations and limitations of each approach to inform the approach we take with VIK.

Grafting involves adding electrical circuits that conform to the surface of a 3D object. Grafting is handy for adding low-profile electronics to existing objects. Prior work has explored techniques such as mounting electronic modules [98], attaching flexible circuit boards [185], adding conformable copper traces [186], sewing in conductive threads [39], and patterning with conductive inks [74], [176], [183]. Overall, grafting approaches can afford close coupling of the electronics with the final device form but often lack discretion and durability. Additionally, these grafts are usually bespoke to a specific form, and changing the reform requires producing new grafts.

Embedding is when the electrical sensing components are integrated within the volume of the objects. Researchers have explored a variety of techniques, such as multi-material printing with conductive material [70], [23], [55], [61], [77], [133]. A common trait of embedding approaches is the co-fabrication of form in tandem with electronics, which allows a more seamless electrical integration. However, iteration and repair are difficult without damaging the larger structure, and complete refabrication is usually easier. Rapid prototyping with these approaches also requires maintained access to digital fabrication equipment, restricting where and when interaction can occur.

Ensemble Configuration, the third and final approach, employs electromechanical building blocks that can be (re)assembled into various forms. Blinky Blocks [102], a foundational example of ensemble assembly of digital materials, are cheap building blocks that can be freely assembled into 3D structures with magnetic interconnects. Commercially available ensemble configuration kits exist today, such as littleBits [32] and LEGO Mindstorm [20], but none support the facile creation of human-scale structures. Since ensemble configuration focuses on the rapid reconfiguration of inexpensive building blocks, these approaches prioritize inter-module connections that are fast and easy to (un)make but rarely robust. This can still support the construction of human-scale assemblies, like those seen in Strawctures [180], but with limited load-bearing capabilities. Overcoming this limitation is a central challenge to producing large and sturdy structures, and doing so would allow material to be reused instead of consumed by expensive, time-intensive fabrication processes that disincentivize iteration.

2.3.3 Rapid Prototyping of Large & Sturdy Structures

While rapid prototyping at the desktop-scale is fast and cheap, as scale increases, it becomes increasingly slow and expensive. As the size of a prototype increases, so do the cost of materials, the time needed to fabricate, and the need for structural integrity. As a result, several works in HCI have proposed tools to ameliorate the misalignment between rapid prototyping and building large structures. *ProtoPiper* [21] approaches this challenge by creating structures out of tape tubes that are lightweight, cheap to iterate, and quick to form self-supporting structures. However, these artifacts have minimal load-bearing support.

In *Forte* [45], the authors develop a generative design tool that offers material-optimized designs based on the expected loads and desired form, but iteration requires complete refabrication. *Kyub* [31] employs closed-box designs to create sturdy objects. This system allows for rapidly designing large objects with a Minecraft-inspired "boxel" based design tool, which allows non-experts to quickly design sturdy laser-cut objects. However, the work does not optimize the usage and reusability of the fabrication material, making iterations expensive due to the consumption of sturdy materials.

TrussFab [105] uses plastic bottles and 3D-printed joints to create reconfigurable load-bearing structures. The user defines a target geometry, and an accompanying design tool automatically generates a corresponding truss structure and performs structural analysis to ensure suitable strength for the specific application. *TrussFormer* [103] adds pneumatic elements to this system for kinetic structures. The goal of these works is to empower non-experts to design large, structurally sound objects that can be straightforwardly assembled, and using water bottles helps keep costs low. However, the 3D printed joints are bespoke, making them difficult to re-use for different geometries, and the need for a 3D printer may be limiting (getting parts printed via a service results in part costs on the order of $10-$30 per part, which will quickly become prohibitive for large structures). This work is extended to steel tubes in [104], which, though much more structurally sound than bottles, necessitates significant metal-work abilities to recreate. All of these are purely mechanical systems as well— though *TrussFormer* uses actuators, the design tool and system are for designing motion, not for designing electrical routing.

As perhaps the closest related work, *Foxels* [141] creates smart human-scale furniture with reconfigurable modules to introduce the benefits of modularity and flexibility to human-scale design. Still, the work has several areas that could be improved. The first is that Foxels are intended entirely for furniture and prioritize aesthetics over cost. Foxels are also not easily available as the design files are not freely available, and replication would require non-trivial amounts of in-house fabrication. The second is assembling Foxel structures requires 3-6 people, which limits the ability for single users to engage with the system. Third, and most importantly for load-bearing structures, the modules presented are only intended for compressive forces, limiting the achievable geometries as tensile and bending loads are not well supported.

2.3.4 Overview of Related Works & VIK Guiding Principles

With VIK, we introduce a construction system that lowers the barrier of entry for the iterative prototyping of big & sturdy things, both on a financial and skill level. At the same time, VIK preserves all of the freedom and customizability that come with electrical prototyping kits like Arduino but with the ability to be reliably and reversibly integrated into the overall structure. Through the related works, we find three overarching tradeoffs that VIK aims to resolve collectively. Thus, we use these challenges to define VIK's guiding principles: Prototyping vs. Big and Sturdy, Availability, and Electrical Flexibility vs. Structural Coupling.

Prototyping vs. Big and Sturdy: We define prototyping as the rapid and cheap iteration of a design. Based on prior work, we assume that an ideal prototyping system affords reconfigurability— the ability to easily make reversible connections with components with no special tools. We define big as human-scale, and sturdy as having load-bearing capabilities similar to furniture, able to support both compressive and tensile forces like those shown in Kyub [31]. Additionally, big & sturdy structures need to be safe. Thus, our system must provide a built-in workflow for non-experts to quickly evaluate the stability and safety factor of a given configuration.

Availability: Availability refers to the ability to immediately access the system with minimum labor, cost, fabrication tools, and custom processes required. Availability also relates to the educational overhead needed to use the system. A characteristic example of high availability is LEGO Mindstorm [20], [2], which can be readily purchased, leverages common programming frameworks, and provides extensive education support online.

For a general-purpose prototyping platform, it is important to have both. Few will reproduce a fully remixable system that requires fabricating many components and hundreds of parts, especially if the designs are not openly available. On the other hand, a commercially available system that only supports a small number of proprietary modules will inevitably fall short of some users' needs. While a modular set of discrete parts can never encompass all of the things a user may want to create, it should do its best to encompass all the basic needs so that customization is not essential to reach a working prototype. For a VIK system, we prioritize two areas of customization: form and electronics. For electronics, a variety of I/O devices and microcontrollers must be physically and electrically easy to integrate into the structure and network with other devices. For form, the building system should be able to produce a variety of mechanical skeletons that can then be augmented with curved and sloped profiles that can be quickly produced.

Electrical Flexibility vs. Structural Coupling: Electrical flexibility is the ability to, at any point in the prototyping process, support easy configuration and reconfiguration of various electronic modules, sensors, inputs, and custom PCBs. Close structural coupling is the seamless and robust integration of electronics into the structure that avoids single-use adhesives and long-spanning wires that are precarious and messy. Electronics should be embedded into the structure when possible instead of grafted onto the surface to keep them secure and hidden.

The contribution of this work lies in addressing these three guiding principles. Because of the difficulty in reconciling these at times opposing priorities, most of the existing literature focuses on either only one of the guidelines or rather makes a tradeoff between two values (e.g. disregarding electrical flexibility in favor of clean structural coupling. In this work, we aim to synthesize the contributions of prior work to address all three principles and increase options for human-scale electrically integrated mechanical prototyping.

2.4 VIK Overview and Fabrication

VIK is an inexpensive and reconfigurable assembly-based building system that enables the easy construction of large and sturdy structures with integrated electronics: adding each voxel not only builds out the mechanical structure but simultaneously builds out the electrical wiring harness.

The voxel design in VIK abstracts both the structural and electrical engineering knowledge so that the end user can focus on developing interactive high-performance structures without needing significant expertise in either of these fields. Similarly, we designed the electrical system— and how it interfaces with the mechanical system— to be straightforward and error-proof. In contrast to many other mechanical and electrical platforms, VIK minimizes the necessary digital fabrication expertise and access. VIK primarily uses cheap, recyclable components ordered from a PCB vendor. Because VIK is mechanically and electrically reconfigurable, iterating large-scale structures does not result in wasted material. And, even if a user *does not require* the electronic functionality, VIK is a cheap and accessible tool for building lightweight, high-strength structures.

We provide all the files necessary for replication in the Supplementary Materials.

2.4.1 Voxel Mechanical Structure

VIK uses reinforced cuboctahedra, or cuboct, as the base unit cell, as shown in Figure 3B. The cuboct geometry is chosen for its optimal stiffness-to-weight properties [46], while the face-connected geometry is used for easy assembly [91]. Because the cuboct lattice only achieves its optimal mechanical properties when many unit cells are tiled together [72], we add cross-bracing to each face to make the resulting voxel stiffer and stronger.

A single voxel under compression supported a maximum load of 2243.02 N, corresponding to the force exerted by 228 kg weight at the Earth's surface, e.g., the weight of an average upright piano, 3.5 adults, or 2000 times the voxel's own weight. Under tension, it supports a maximum load of 1198 N, or approximately one thousand times its own weight. The voxel has a modulus of 22.58 MPa at a lattice density of 30.756 kg/mˆ3. This is competitive with prior research on truss-based architected materials, such as the high-performance micro-lattices described in [184], and outperforms the nearest mass-manufactured discrete approaches [91], [161] by an order of magnitude. Similarly, a three-voxel beam (including

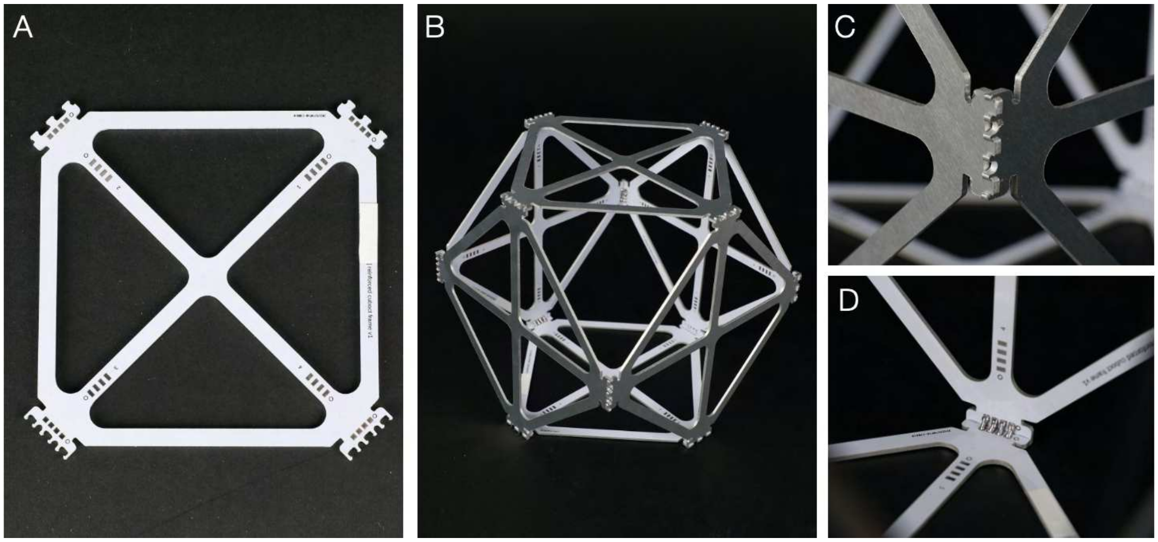

Figure 2-3: A) Top view of a PCB voxel face. B) An assembled voxel. The voxel is a reinforced cuboctahedron unit cell made from six aluminum PCB faces. These are made with small snap-fit features at the corners, as shown in C), to mechanically connect faces. Faces are then soldered, as shown in D), to bridge the electrical connections.

clips) under bending supports a maximum load of 1037 N. Taken together, this means that one voxel could support a ~910-voxel long beam in any orientation, and under compression only, could support up to 2000 voxels. Complete details of the mechanical testing setup and results are available in the Supplementary Materials.

We assemble the voxels from 1.6 mm thick aluminum PCB faces. The assembled voxels fill in a 150 x 150 x 150 mm bounding box— this size is chosen as appropriate for human hands to fit into. Aluminum is chosen as the PCB substrate as it provides good mechanical properties at a cheap cost with minimal design and fabrication effort required from the end user.

Currently, a user could order our design directly from a board house at a low cost (we used JLCPCB, which priced the basic frame at approx. $0.70 per face at an order quantity of 250 PCBs), requiring no fabrication equipment aside from a soldering iron. It is worth noting that it is orders of magnitude cheaper to purchase these faces as aluminum PCBs than as custom CNC profiles or as outsourced 3D-printed parts. Even without the added benefit of included PCB routing, the aluminum PCB voxels represent a cost-effective strategy for achieving large-scale, high-performance structures.

2.4.2 Basic Voxel Frames and Connector Systems

VIK uses two types of structural PCB face: a basic frame that routes power and signal but carries no electrical components, as shown in Figure 2-3A, and a microcontroller frame,

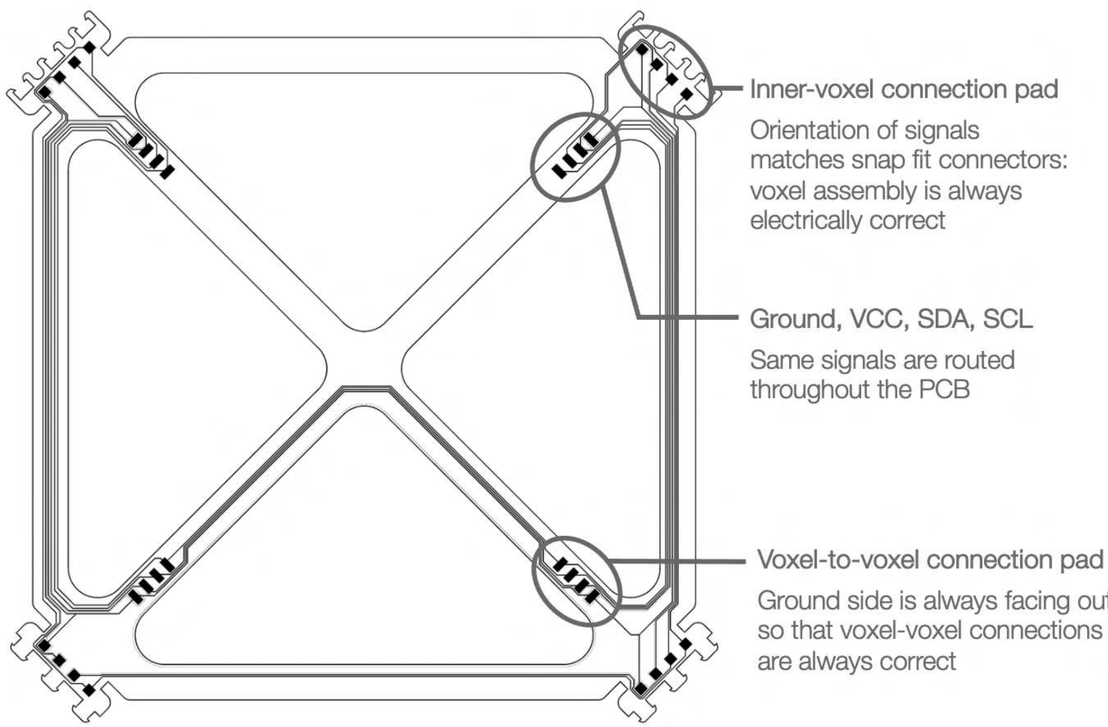

Figure 2-4: Diagram of the PCB routing. Four signals are routed throughout the structure: Ground, power, and two data lines. These signals are exposed at the corners for inner-voxel connections and on the beams for between-voxel connections.

which integrates a microcontroller directly into the lattice. VIK additionally provides a small library of traditional PCB attachment boards, which mount directly to the basic frame (further discussed in the following section). A snap-fit joint makes the mechanical connections between faces within a voxel. Figure 2-3C shows a connected voxel. We make the inner-voxel electrical connections by bridging exposed pads at the corner of each voxel with solder, as shown in Figure 2-3D, similar to the system used in [42] or [165]. Although soldering every corner in a voxel is unnecessary, it reinforces joints and provides failure-avoidant electrical redundancy. Each voxel has twelve solderable joints; it is only necessary to solder five to achieve full electrical connectivity within a voxel.

In this implementation, VIK routes four signals: power, ground, and two data lines, which we use as the data and clock lines for I2C [4], a common and simple communications protocol for microcontrollers widely used in, e.g., SparkFun and Adafruit sensor boards. Basic frames are designed with multiple breakout pads with these four lines to enable easy interfacing with custom boards (such as the microcontroller attachment boards in this project) or with commercial sensors or development boards, which often include I2C as an interface option. Figure 2-4 shows the routing of the basic frame, showing the joining of all the connection pads within a voxel frame. The snap-fit connectors orient the four inner-voxel connection pads at the corners of the voxels— this means that it is impossible to connect two voxel faces with the wrong electrical polarity. The voxel-voxel connection pads on the inner beams of the voxels are all oriented with the same polarity, which means we can make voxel-to-voxel connections from any side or orientation of the voxel face.

We make voxel-to-voxel connections using off-the-shelf connectors (see Figure 2-5). We specifically use Qwiic connectors due to their compatibility with many popular electronics prototyping systems (Sparkfun, for example, carries 320 products tagged with "Qwiic"), though any other 1 mm pitch 4-pin connector may be substituted. The connectors have a wider range of deformation than the lattice, so under loading, they will only fail after the voxels have already mechanically failed. The separation of the electrical connector and mechanical connector also enables more advanced users to route multiple I2C networks in a single structure or to more easily adapt the existing physical traces for their preferred network.

Voxel-to-voxel mechanical connections are made using a snap-fit clip that latches into indexing features on the voxel faces, as shown in Figure 2-6. The clips are also made from aluminum PCB via a board house, at <$0.10 per part. We explored two clip configurations: a 2-prong and a 3-prong version. We recommend users use the 3-prong version for added stability, though we include an image of the 2-prong version for reference as we use it throughout this paper.

2.4.3 Integrating Microcontrollers

A structure made using VIK will likely consist of basic frames with microcontrollers placed into the structure as needed. We currently implement multiple microcontroller types so that users can choose what they are most comfortable with. The first set is built around

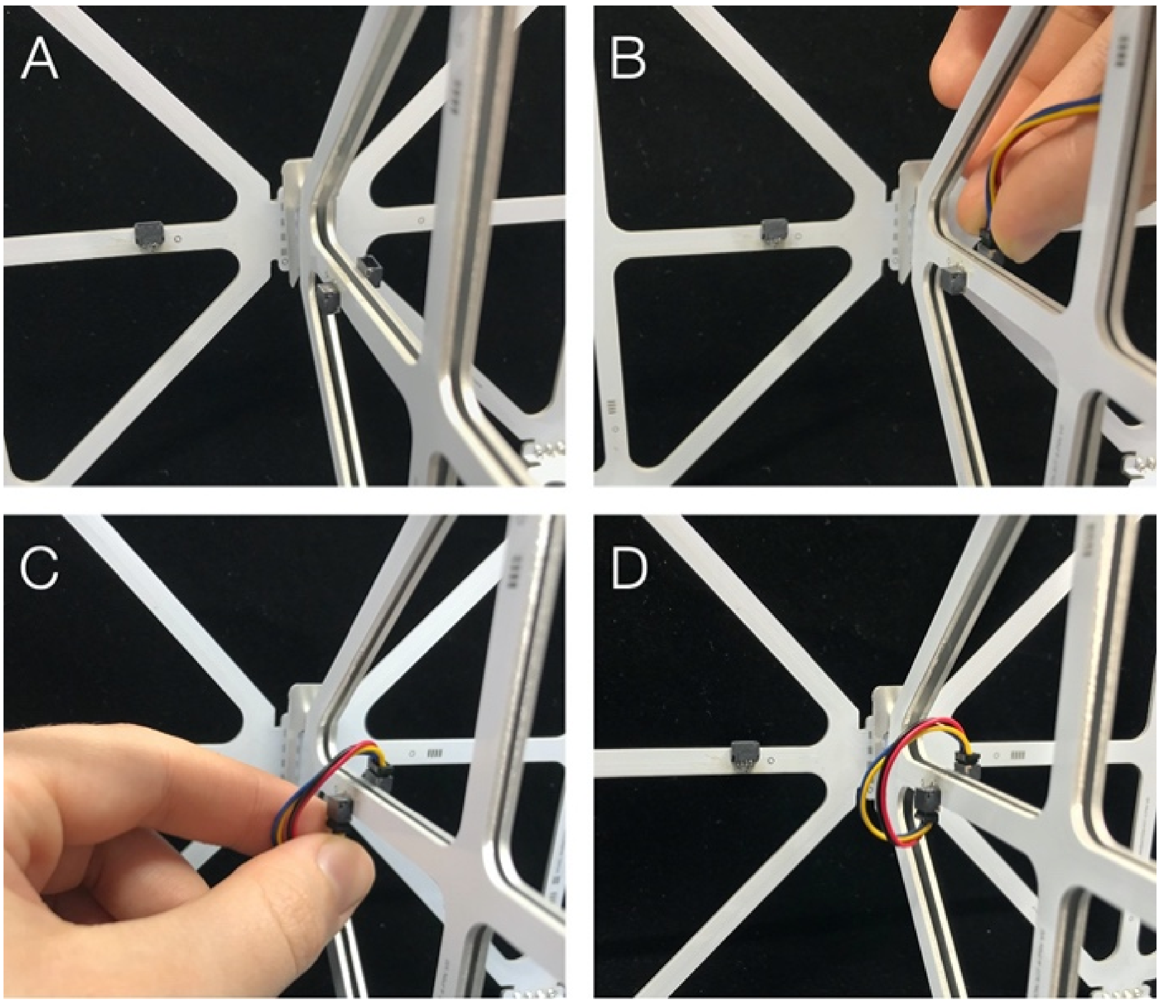

Figure 2-5: Installation of the Qwiic connector. 1) shows the two voxel faces to be connected, with the connector pre-soldered in the same position on each face (the connector we have chosen is polarized, so the connectors must be installed consistently for the cable to plug in). 2-3) the cable is plugged in, and 4) the resultant connection.

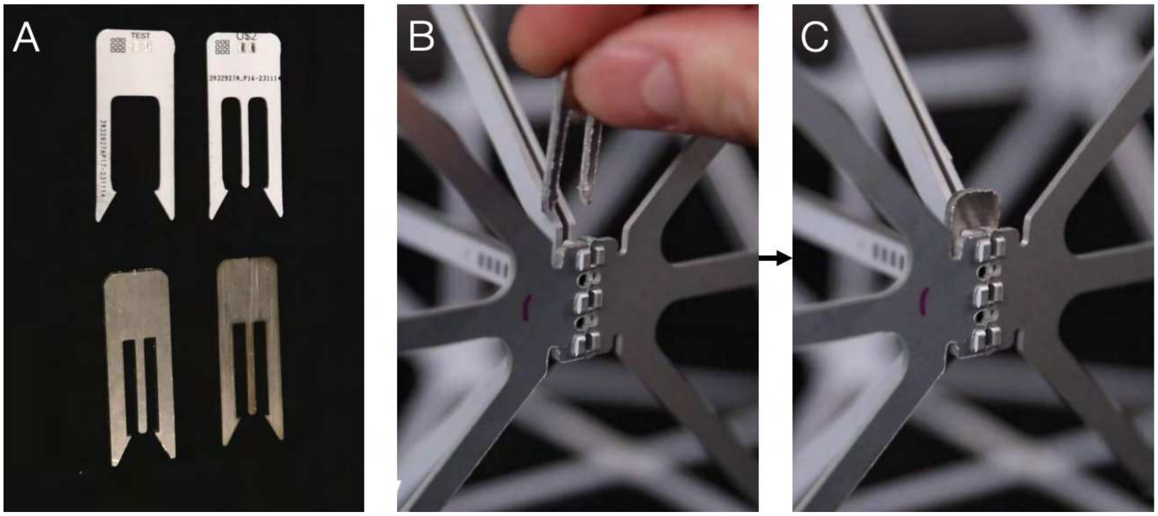

Figure 2-6: A) Two-prong clip and three-prong clip examples. B and C illustrate the installation process of the clips.

the ATtiny1626 microcontroller, which we chose as it is low-cost, robust, and compatible with the Arduino programming environment, making it easier to use. This is provided as both a frame-integrated board and as a separate attachment PCB (see Figure 2-7).

We built the second family of microcontroller boards around ESP32 microcontrollers for added WiFi and Bluetooth capabilities (see Figure 2-7). The current VIK system architecture typically uses one ESP32 microcontroller as the primary controller of the I2C bus, with ATtiny boards as targets (though the current hardware is compatible with some alternate networking architectures). ATtinys would act as interfaces to input and output devices, with the ESP32 determining overall behavior. This architecture enables easier reprogramming, as users can change the system behavior by modifying only the ESP32's code instead of all of the ATtinys. This enables easy interfacing with external applications: the ESP32 could be controlled via a smartphone over Bluetooth or accessed via a website, given its WiFi capabilities. In the Supplementary Materials, we provide the schematic and board files for all the custom development microcontroller boards used in VIK so that users can easily order them from a board house and assemble the PCBs.

For users who are less comfortable with microcontrollers, we additionally implement boards based on the Seeed Studio XIAO development boards, which are small-form-factor and require minimal intervention to interface with VIK. Development boards with built-in Qwiic connector interfaces, such as the Sparkfun Artemis Nano, may also be directly integrated into VIK with no soldering, which we show in Section 6.

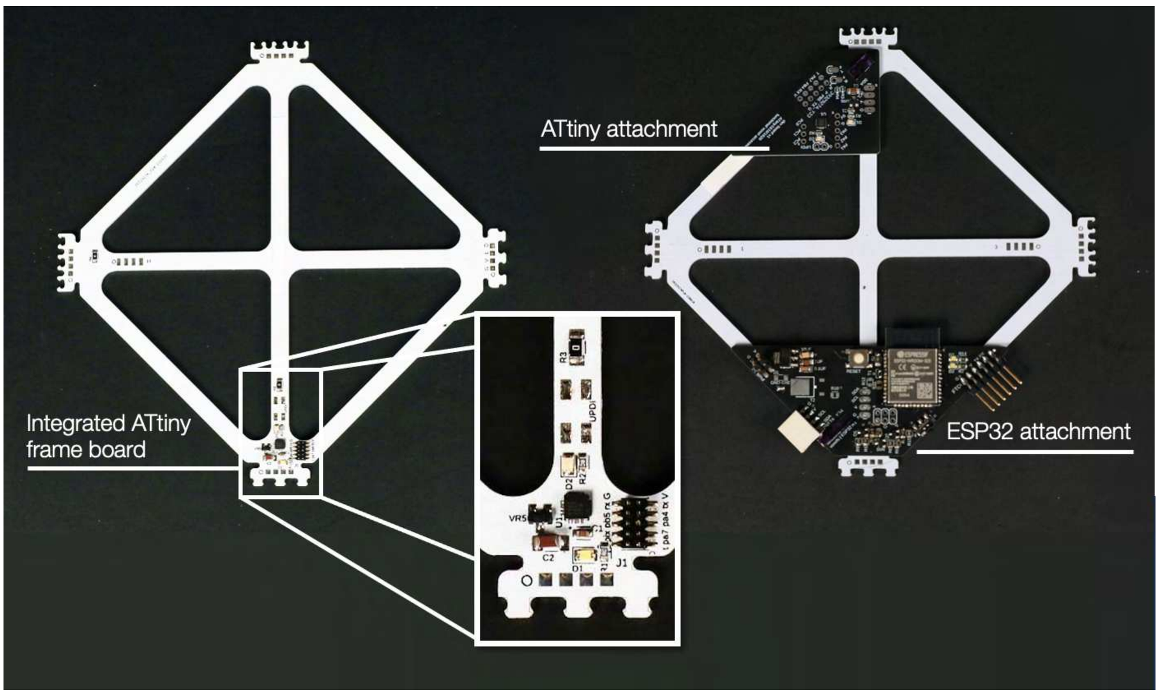

Figure 2-7: (Left) ATtiny microcontroller frame. The inset provides a zoomed-in view of the integrated microcontroller on the frame. (Right) A basic frame with an ATtiny attachment PCB installed on top and an ESP32 attachment installed on the bottom.

2.4.4 Electronic Modules

Input and output devices are added to the VIK system to build out electronic modules and customize the system's operation to the end user's desire. We designed the microcontroller frames and the basic frame to facilitate the inclusion of almost any input or output device an end user may require. Sensors, actuators, or microcontroller development boards with I2C interfaces directly integrate into the basic frame of VIK. For devices or systems without these interfaces, we have developed microcontroller board designs to operate de facto as an I2C device in the lattice.

We have explored a set of both self-configured and commercially available I/O devices. Primarily, we have focused on implementing our own version of capacitive force, touch, and proximity sensing as inputs to our system and used LEDs, speakers, and vibration motors as outputs, while additionally demonstrating a selection of Sparkfun and Adafruit Qwiic modules on VIK For capacitive sensing, we use either a two-plate system for force sensing or a single-electrode version of a proximity and touch system, as in [1]. We make the electrodes out of copper tape and focus on this one input style as it is simple and cheap to implement yet enables a wide range of tactile and non-contact interactions. For our outputs, we tested LEDs, NeoPixel LEDs, speakers, and vibration motors, though we can also trivially drive other similar systems (e.g., DC motors, buzzers, hobby servos). The Applications section implements several examples of these systems.

2.5 Design Tool Overview

2.5.1 Design and Assembly

We developed a physical computing design tool: an online integrated tool where the users design a structure and simulate the loads/deflection of the structure simultaneously, based on the tool introduced in [64]. The goal of this is to enable users without engineering backgrounds to rapidly validate the structural stability of their designs, as well as to provide a fast way to design in a novice-friendly Minecraft-like environment. The tool has two main windows (see Figure 2-8). In the first "Assembly" window, the user places the voxels on a predefined grid the same size as the voxel. The cursor automatically snaps so voxels can only be placed on the base grid or attached to previously placed voxels. In the background, the assembly sequence is saved, and one can later replay this assembly sequence to help inform the assembly process.

When the user right-clicks on any of the voxels in the "Assembly" window, a radial menu appears, and one can select, delete, or change the type of the voxel placed. There are five types: rigid/base voxels, microcontroller voxels, sensing voxels, display voxels, and load voxels (where loads/forces are applied). When right-clicking on the background of the "Assembly" window, three buttons appear: one can either save, upload the design, or start a static simulation to calculate the deflection of the structure. To save and upload

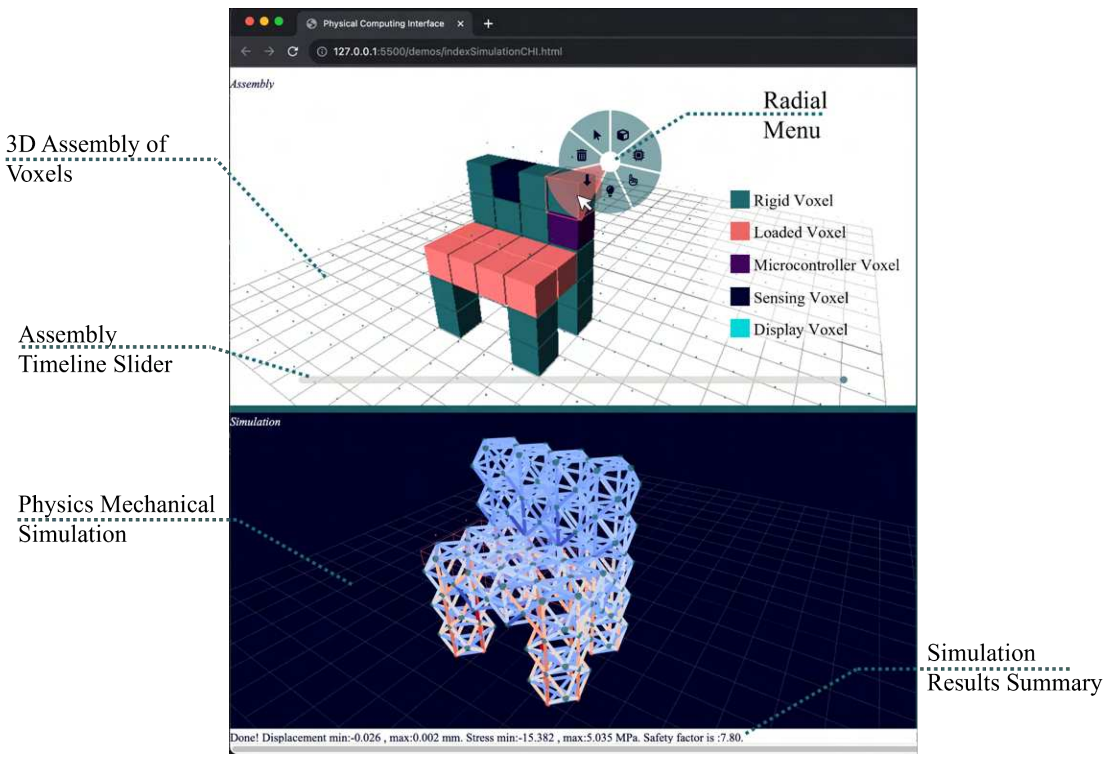

Figure 2-8: Design tool overview. The design tool has two windows: Assembly and Simulation. Voxels can be assembled in the Assembly window, and mechanical response to applied loads can be simulated in the Simulation window along with the summarized results, including safety factor.

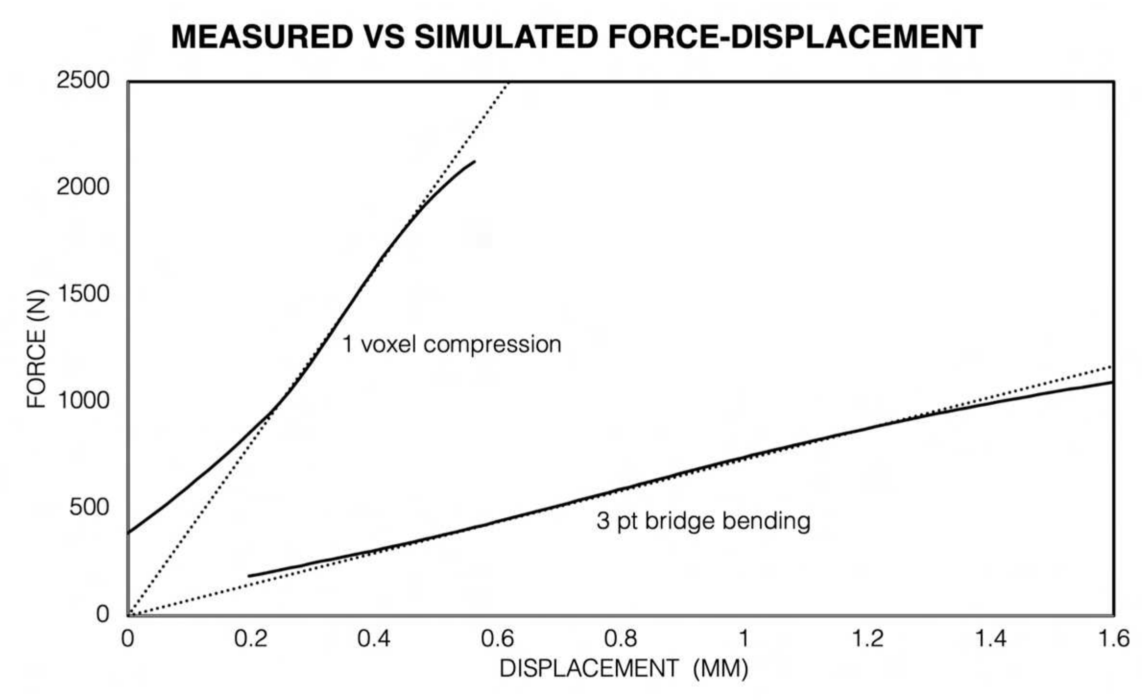

Figure 2-9: Measured vs. simulated results for force-displacement in the elastic range, showing good agreement with the toe region removed.

the design, we use a .json file format to capture all the voxels added in the design session, including their properties and placement order. We additionally export a suggested BOM, in terms of PCB face count and type to facilitate easy part ordering.

2.5.2 Structural Mechanics Simulation

Our goal for the structural mechanic simulation is to provide fast and reliable feedback to the designer— unlike fully meshed solid FEA models, which require expertise to set up and frequently fail for complex structures like voxels, our system returns results in seconds. We developed a static linear finite element analysis (FEA) implementation to perform reduced-order beam model simulation using Bernoulli-Euler beam elements for the simulation engine used to model the deflection of the structure under self-load and applied external load. The simulation discretizes the voxel structure into node and beam elements. Note that the clips between voxels are modeled as beams, with their constitutive properties set based on measured data (see the Supplementary Materials for more details), but because of their short length, they are not easily visible. Each node has 6 degrees of freedom: three rotational degrees of freedom and three traditional translational degrees. We use the Bernoulli-Euler beam theory to prescribe the biaxial bending, transverse shear, and axial stretching that each beam experiences.

In [91], a study compares the results of reduced-order beam FEA simulations in comparison to fully detailed meshed voxels simulated using solid FEA models, showing that the simulation results are in good agreement with each other and the experimental data. However, the fully meshed FEA model required approximately three orders of magnitude more elements than the corresponding beam model. Since the purpose of the design tool is to give fast and instant feedback on design decisions, we chose to use the beam model as it achieves the best balance between simulation speed and accuracy. We also compare our simulated elastic response against the measured response for a few voxel geometries, showing good agreement for both single voxel compression and 5-voxel 3-point bridge bending (see Figure 2-9 and the Supplementary Materials for mechanical testing details).

After the user runs the simulation, an animation of the structure's deflection (exaggerated 10x) appears, and the beams are colored based on their stress value to identify potential points of failure. Finally, a summary of the results (minimum and maximum displacement, minimum and maximum stress values, and safety factor) is calculated and displayed at the bottom footer.

The user can make informed design decisions by having instant feedback and visualization of the structural assembly and simulation in a simple integrated design tool without going back and forth between different interfaces. Importantly, the user does not need to understand any of what this section just described. Users assemble voxels together like in Minecraft, and the simulation window helps users understand the flow of mechanical forces in their system without experience in this area.

2.6 Slopes and Curves

Many of the human-scale structures we interact with include curved forms and soft padding that clad an object's 'mechanical skeleton'. Since the forms made from the voxels on their own are rigid and cubic, we designed a fast, inexpensive, and machine free way to make curved profile modules of varying stiffness using PCBs to make molds for expanding foams. The goal of this is to provide users without access to digital fabrication equipment or CAD background a fast, easy, and cheap way of producing a small library of curved/interpolating shapes to augment VIK with.

The exterior of the mold is made from slot-fit rigid PCBs with a templated flex-PCB, which can be cut and folded into preset shapes. The total cost of the PCB/flex-mold is ~$14 and can be ordered along with the VIK faces (see Figure 2-10). This approach may seem circuitous, but importantly, it does not require digital fabrication equipment, is still cheaper than ordering custom 3DP or laser cut parts, and is relatively fast: the two foams we used, Smooth-on FlexFoam for soft parts and FoamIt for rigid parts, solidify and can be handled after 30 minutes and fully cure in 2 hours. This means with ten molds, we could produce 20 shapes in one hour and would be fully deployable in 2.5 hours. The system is also reconfigurable, much like the voxels (see Section 6.4 for reference). We provide a detailed instruction walkthrough for this in the Supplementary Materials. For more advanced users,

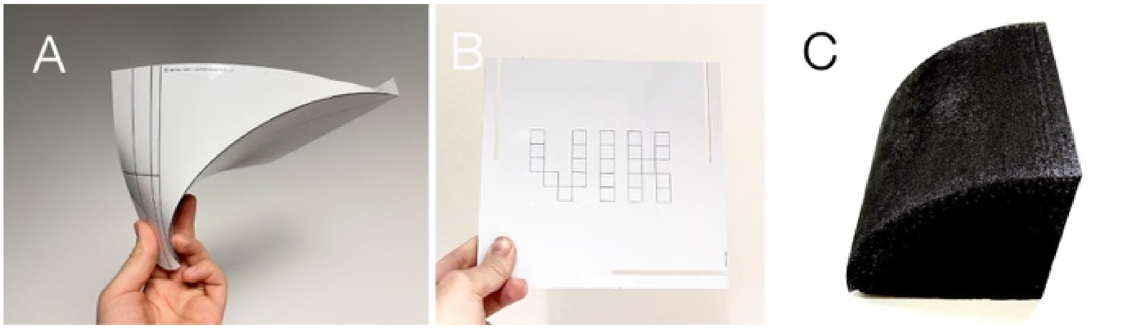

Figure 2-10: A) A curved-crease folded flex PCB molding insert. B) One side of the rigid PCB molding base. C) A resultant soft-foam curved block.

we additionally provide the base 3D models in the Supplementary Materials, which may be augmented to provide more customized molding surfaces.

Though there are limitations to this approach, it is still a fast and easy way to generate more accurate looks-like and feels-like prototypes with VIK, which is our priority. We additionally provide the designs for the other surfaces we used, so users with fabrication equipment access or CAD experience can build off these methods as well.

2.7 Applications

In this section, we demonstrate how VIK enables the reconfiguration of a standard set of voxels to create various interactive devices. Here, we revisit our hypothetical designer, Victor, who has a library of VIK parts and wants to build some interactive structures. Each demo object is a large, electronically integrated system that would be difficult to produce via standard digital fabrication tools. VIK enables Victor to (almost) trivially assemble these human-scale, load-bearing applications and place electronics into the structure without needing to think about wiring harnesses, cable management, or structural validation, as VIK handles all of this.

Further, VIK enables the complete re-use of the voxels. Starting from a dance floor gamepad, each demo is assembled (and disassembled) from the same set of voxels. This is illustrated in Figure 2-11, where the dance floor becomes a chair and mobile robot, which then combine, are disassembled, and added into a bridge, which is then transformed into a lounge chair.

2.7.1 Party Dance Floor

In the first example, Victor is hosting a party and wants a fun activity for the guests—a Dance Dance Revolution-style game. But, they're daunted by the task of ensuring that

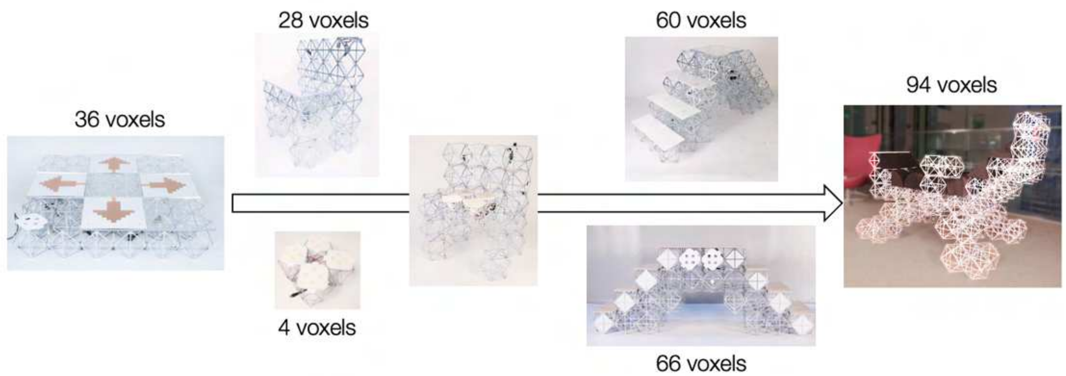

Figure 2-11: Flow of voxels used in the example applications.

what they build is safe— a 70 kg person jumping on the structure repeatedly exerts forces in the range of 1700 N— and that the electronics system can survive these loads, and that their friend who's really into Dance Dance approves of the physical feel of the dance floor. Additionally, they don't have access to any shop tools, let alone digital fabrication machines, which puts most existing solutions such as [79], [88] out of reach. With VIK, they can quickly implement a capacitive sensor-based dance pad using a 6x6 bed of voxels, using only a soldering iron, scissors, and pliers (see Figure 2-12).

Victor first assembles the 6x6 voxel grid with a few friends before outfitting it with electronics. After doing a little research online, Victor determines the easiest solution will be capacitive sensing arrows and USB emulation so that they can play the game with an existing online dance floor game— this is all functionality VIK supports. Four active dance direction sensors using capacitive sensing are implemented by plugging in VIK attachment ATtiny boards to the voxel grid. The top sensor surface is formed from acrylic panels that sandwich copper tape to create durable pressure sensors. The sensing boards, which additionally control NeoPixel LEDs, are networked with a VIK ESP32 board, which emulates a USB keyboard for compatibility with popular open-source dance games (in our case, we adapted a JavaScript web app from [53]). The dance floor logs approximately two person-hours of total use and shows no signs of mechanical or electrical damage, indicating good fatigue performance. If we assume a person dances at 1 Hz and exerts 1.5kN with each move, evenly distributed over one of the plates, then each of the loaded voxels goes through ~480 load cycles of 375 N. As this is only ~17% of a voxel's total loading capacity, it follows that we would not expect fatigue failure.

2.7.2 Interactive Chair Version 1

Once the party is over, the dance floor is no longer needed. Instead of having to forever store an awkwardly large piece of equipment (or throw it out, wasting the materials, cost,

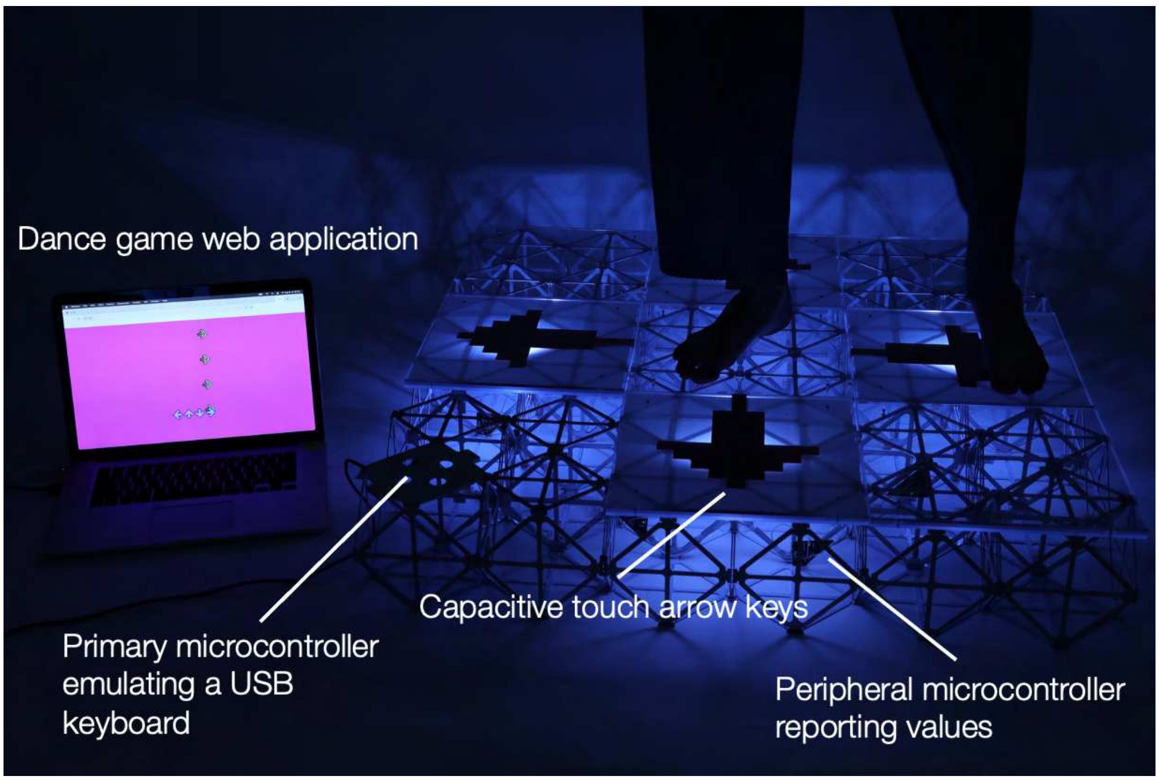

Figure 2-12: Dance floor with subcomponents labeled.

and effort), Victor disassembles the dance floor without compromising mechanical stability or electrical functionality.

Victor then gets a new hobby: experimental music instruments, and so reconfigures the voxels into a chair that is also a theremin: a Chairemin. Wanting to reuse as many components from the dance floor as possible, Victor first runs through a few designs of the chair in the design tool to help visualize potential sensor locations, as well as to validate the structural performance (see Figure 2-8). Victor finds that for their weight, approximately 80 kg, the chair has a safety factor of 7.8, which is plenty, and sets off to build it. The Chairemin uses two capacitive proximity sensors to detect the user's body while sitting on the chair (see Figure 2-13 for labelled components of the Chairemin). The sensor on the back of the chair controls the volume of a tone sent to a speaker module, and the sensor on the side of the chair controls the frequency. The Chairemin is played by using their hand to control the frequency, which offers finer-grained control, while the volume is controlled by how far back they lean into the back of the chair.

The Chairemin reuses two clear panels from the dance floor to form the seat and reuses two capacitive touch microcontrollers for the audio controls. Note that the Chairemin does not require the acrylic panels to support loads, as shown in Figure 2-14, and is there for comfort.

The combined electrical and mechanical system means that it is trivial to try different configurations for sensor and audio placement so the experience can be tailored for a specific user. To try new sensor locations, we can simply plug the capacitive sensors into any of the voxels in the structure (without needing to cut new wires, de-solder, re-solder, etc., as would be necessary with a standard setup). This makes it easy to adapt the Chairemin for use by people with different body types and easy to quickly iterate different interaction modes.

2.7.3 Mobile Robot to Massage Chair

After transforming the dance floor into the Chairemin, Victor uses the remaining four voxels to create the Slowbot, a vibrating robot for passively sweeping floors at speeds of up to 3 cm/s. Victor already has the electronics for the system made and is just looking for a convenient and lightweight mechanical harness, and so uses the leftover voxels.

The slow bot is assembled using a 3D-printed directional fur (PolyBrush, OPT Industries). The directional fur means that the slow-bot's ground-facing surface has lower friction in the direction of the fur and higher friction opposite to it. This anisotropy biases the movement caused by the vibrating motors to move the robot in one direction.

Ultimately, Victor decides that the Slowbot is indeed too slow and an uncompelling use of their voxel kit. So, they pop the seat of the Chairemin out and replace it with the Slowbot (see Figure 2-15). A quick code change switches the capacitive touch sensors to trigger the vibrational motors instead of the speaker output, taking about 10 minutes total to reconfigure and reprogram the chair.

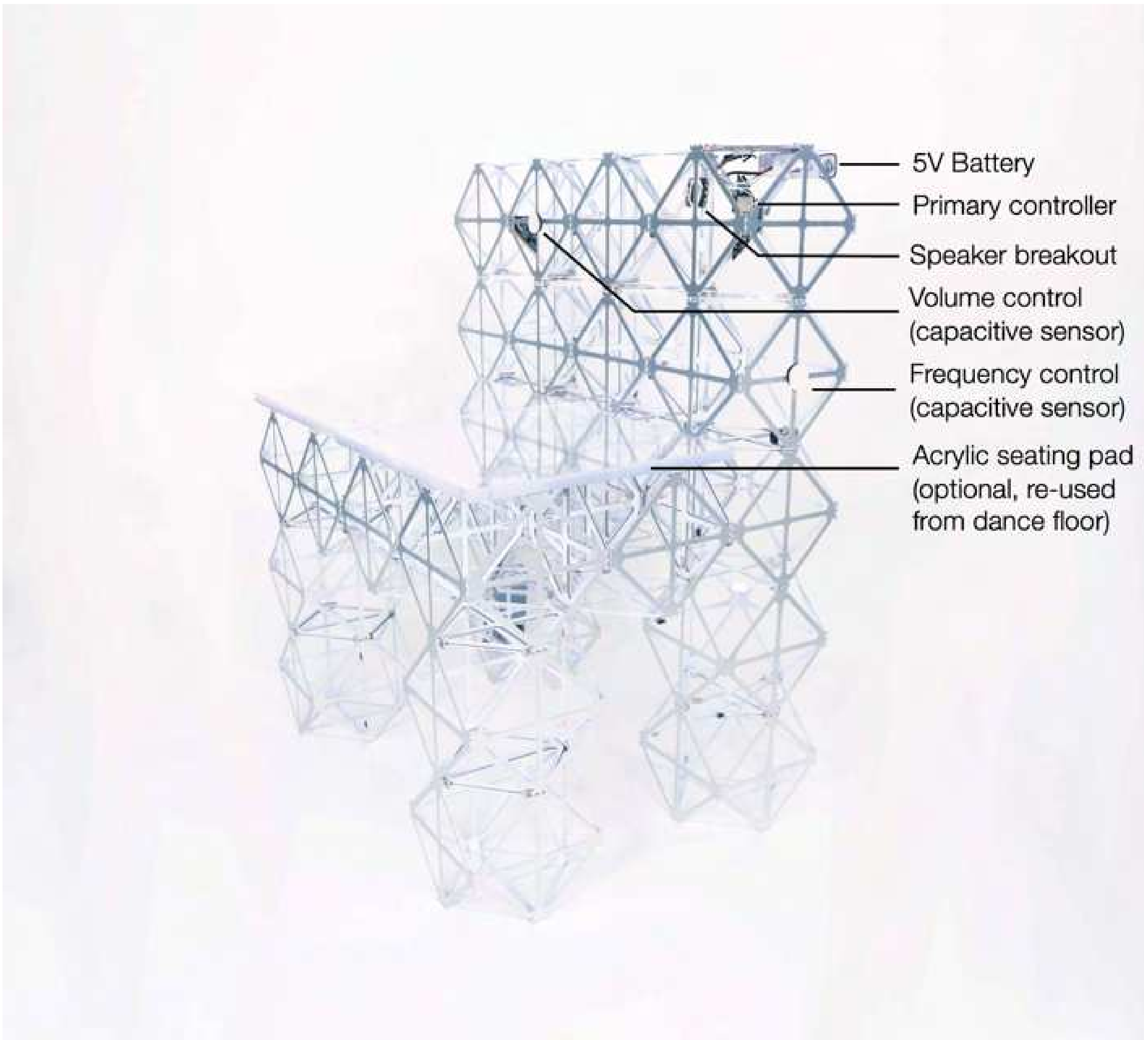

Figure 2-13: Chairemin with labeled components. The primary controller, speaker, and rechargeable battery are all placed within the same voxel in the upper corner of the chair. The capacitive sensing modules are placed one and two voxels away, with the sensing pads placed on the exterior frame of the chair.

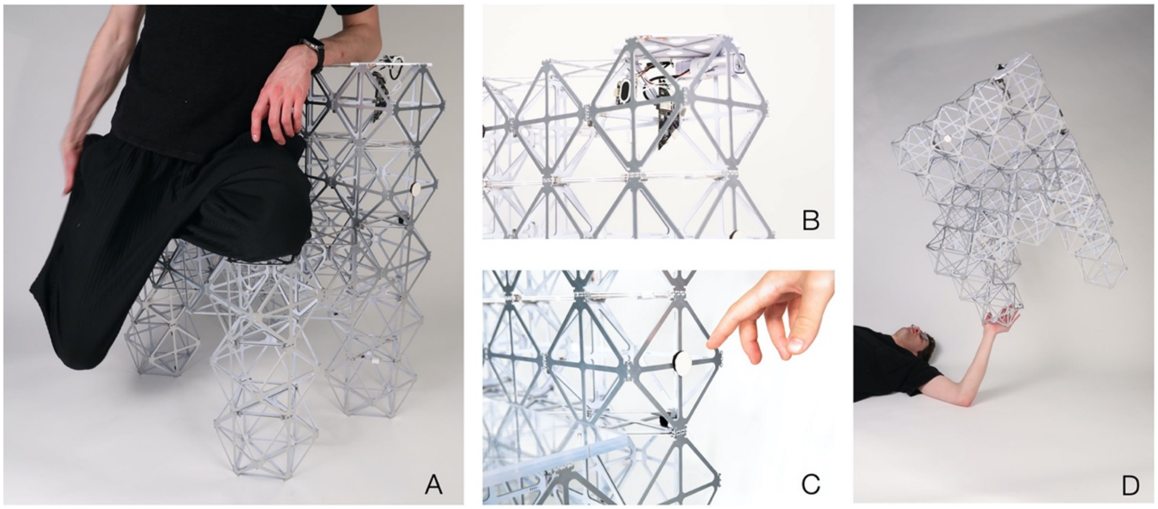

Figure 2-14: A) A person fully supported by the Chairemin. B) Close-up view of the primary module, containing the ESP32 controller, Adafruit speaker board, and 5V battery. C) View of the frequency control capacitive sensor electrode. D) The Chairemin is lightweight.

2.7.4 Bridging Play Structure

Victor then wants to build a play structure for their child and decides to assemble a miniature bridge (termed "abridged bridge") with embedded load sensing, which will allow them to track and test what structure configurations result in the most interactions. Load sensing is implemented via two-plate capacitive distance measurement using microcontroller frames, as shown in Figure 2-16. Voxels are additionally embedded with NeoPixels, and all systems are coordinated via a WiFi-enabled microcontroller and powered via USB 5V battery packs.

The bridge is initially assembled with a 3-voxel length span, as shown in Figure 2-11. In this configuration, Victor tests out different materials for the stepping surfaces and finds, based on the steps logged, that wood panels result in the highest usage. The user also tests the addition of speakers and proximity sensors (as in the Chairemin) to the bridge and finds that this further increases engagement. The user then additionally notes that if the bridge length is increased by one voxel length, as shown in Figure 2-16, then the children can more easily crawl underneath it as well.

VIK enables these systems to be almost trivially tested and added. The reconfiguration of the bridge to span a 3-voxel gap to a 4-voxel gap is a process that only requires pliers and no additional new electrical work. In contrast, via standard fabrication and digital fabrication processes, the increase in bridge size would have necessitated a likely re-design of the system, re-fabrication of the parts, and rewiring of the structure, all of which is a time-consuming and likely wasteful process based on the scale of the structure– with VIK, the process took thirty minutes.

Even though the bridge is made from previously used voxels (from the dance floor and chair), VIK is sturdy enough— mechanically and electrically— to be used in high-load

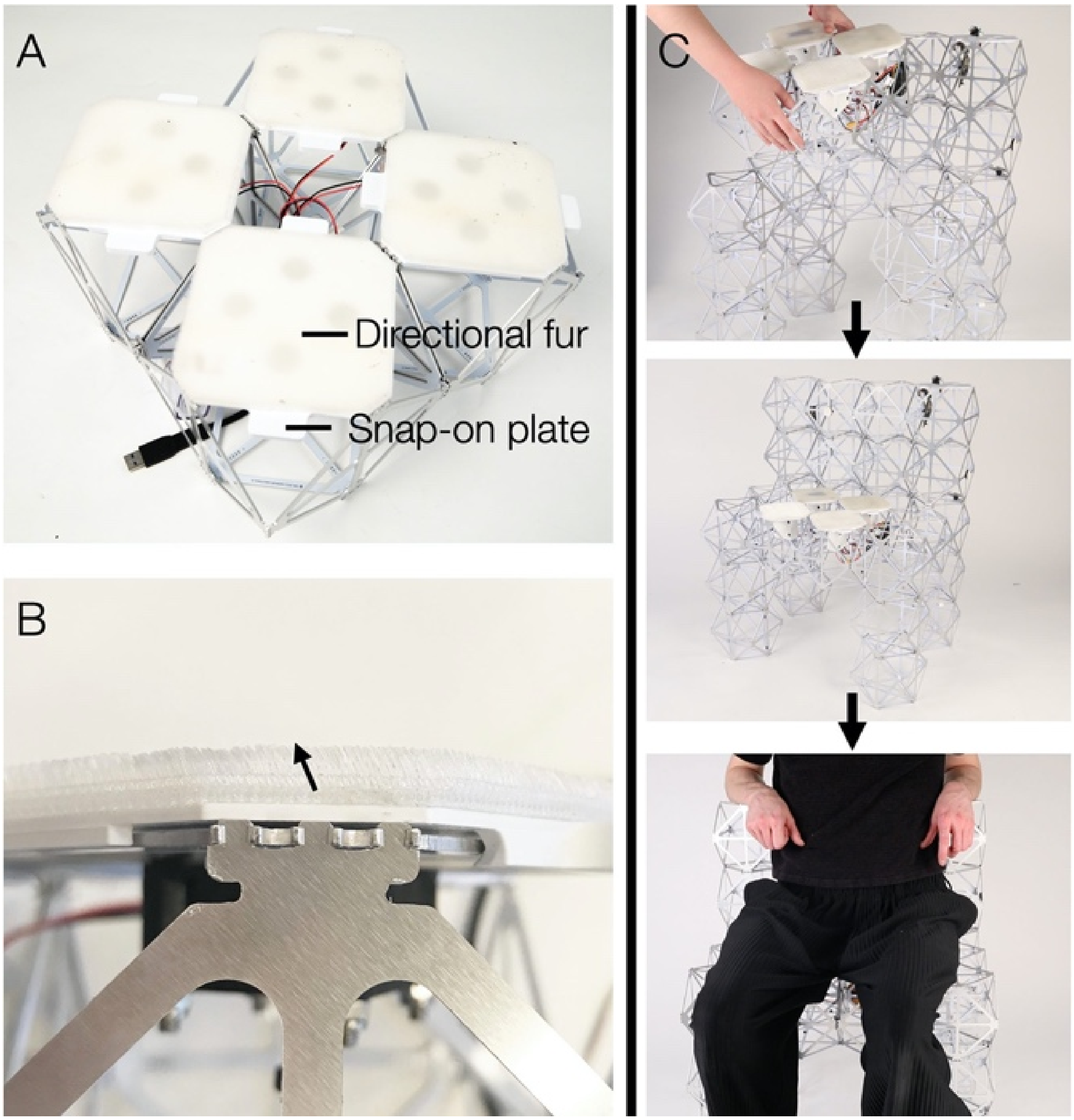

Figure 2-15: Overview of the Slowbot. A) Bottom view of the Slowbot, showing the directional fur modules. B) Close-up view of directional fur, with an arrow indicating its orientation. C) Installing the slowbot into the Chairemin.

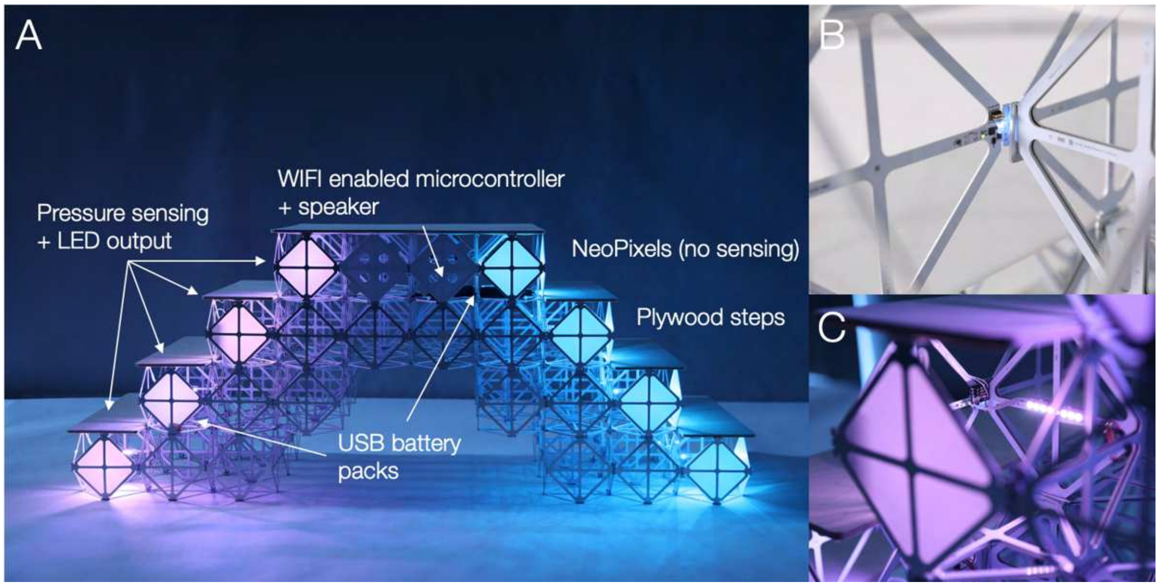

Figure 2-16: A) The assembled bridge with components labeled. B) A close-up view of one of the embedded microcontroller frames. C) The microcontroller frame with a capacitive force sensor and NeoPixels.

applications, taken apart, and reused for new structures. To demonstrate the load-bearing ability of the bridge, we had a person jump up and down on the top surface four times, and we had previously had a person jump on the 3-voxel gap version of the bridge three times. For one of the jumps, we then calculated the average impact force to be 5800 N based on the jump duration, landing duration, and the person's mass, as shown in Figure 2-17. Mechanically and electrically, the bridge withstood multiple jumps, indicating that its ultimate load-bearing capacity is much higher, which aligns with our expectations. We then removed three voxels from the middle of the structure and tested them under compression (see the Supplementary Materials for reference), recording a maximum average load of 2216 $\pm$ 29 N and modulus 24.2 $\pm$ 0.7 MPa, showing no significant difference in strength or stiffness for voxels which at this point had been used in the dance floor, chair, and bridge (details of the mechanical testing are available in the Supplementary Materials).

2.7.5 Interactive Chair Version 2

As time goes on, Victor finds themself reflecting on the Chairemin and wishing they could push the concept of an interactive chair further. The simple design of the Chairemin is overly demure compared to the silliness of the chair's function, inconsistent with the other furniture around, and not soft to sit on. In the second version, Victor sets out to correct this.

Victor first spends twenty minutes iterating chair designs in the design tool so that they can quickly explore a few very different chair concepts and then verify that their preferred

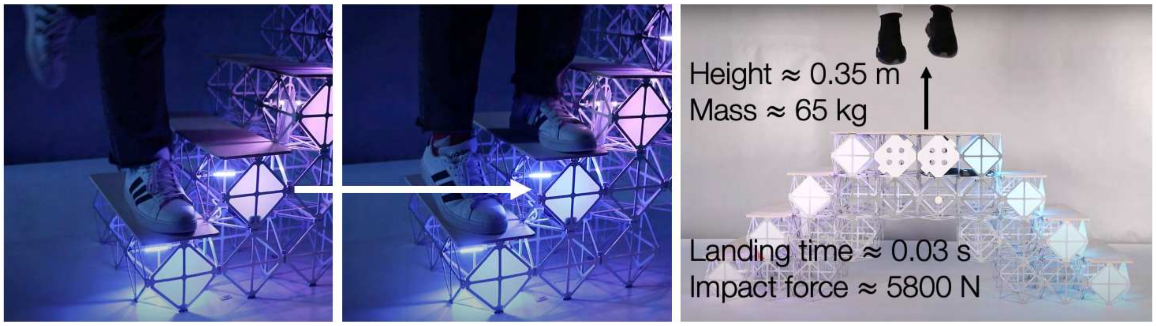

Figure 2-17: The stair lights' step-response behavior and the approximate jump impact force calculation.

cantilevered shape is also sufficiently strong to support their weight. After this, Victor prototypes the chair in situ so they can evaluate how the chair looks and functions where they intend to install it. Victor begins by building out the base and legs and then builds out the backrest and leg rest as separate modules so that they can physically evaluate what looks best and works best. Once they're satisfied with its shape, they outfit it with the modular soft foam shapes (see Section 5 for reference) so that the lounge chair is comfortable for lounging. Because these shapes are also modular, Victor can quickly try a variety of arrangements of curves, slopes, and blocks until they find a configuration they find comfortable (see Figure 2-18).

After this, they outfit the lounge chair entirely with commercially available development boards and modules. Because they are unsure what functionality they ultimately want with the chair, they iterate through different IO modules and physical voxel placements: initially, they install a Sparkfun Qwiic joystick module into a low armrest, which they then later switch out for a Grove Gesture sensor. After using the sensor, they then rebuild the armrest to be more comfortable based on their actual physically tested experience, all of which takes about 10 minutes. Ultimately, they install a Grove Gesture sensor, Sparkfun PC fan controller, Qwiic speakers, and Qwiic LED strips, which are then controlled via a Sparkfun Artemis Nano, which has a Qwiic connector as well, and so can be plugged into the structure (See Figure 2-19). Power is supplied by a variable-voltage USB battery pack.

The lounge chair operates by inputting different gestures over the armrest, which toggle the lights in different settings, as well as control audio over a Bluetooth connection. This is designed so that one can lounge in the chair and place their laptop or phone on the stand opposite them and control it without needing to lean forward for peak comfort (see Figure 2-20).

In these demonstrations, hypothetical Victor can build and rebuild a series of load-bearing human-scale interactive devices using very simple tools, with confidence that the building system handles all of the tricky details that usually come with building big things. The goal of VIK isn't to be the end-all-be-all of building human/furniture-scale objects. Instead, it's to enable the rapid prototyping and iteration of these structures. Currently, prototyping at this scale is much more limited than at the desktop scale, and VIK (hopefully) represents

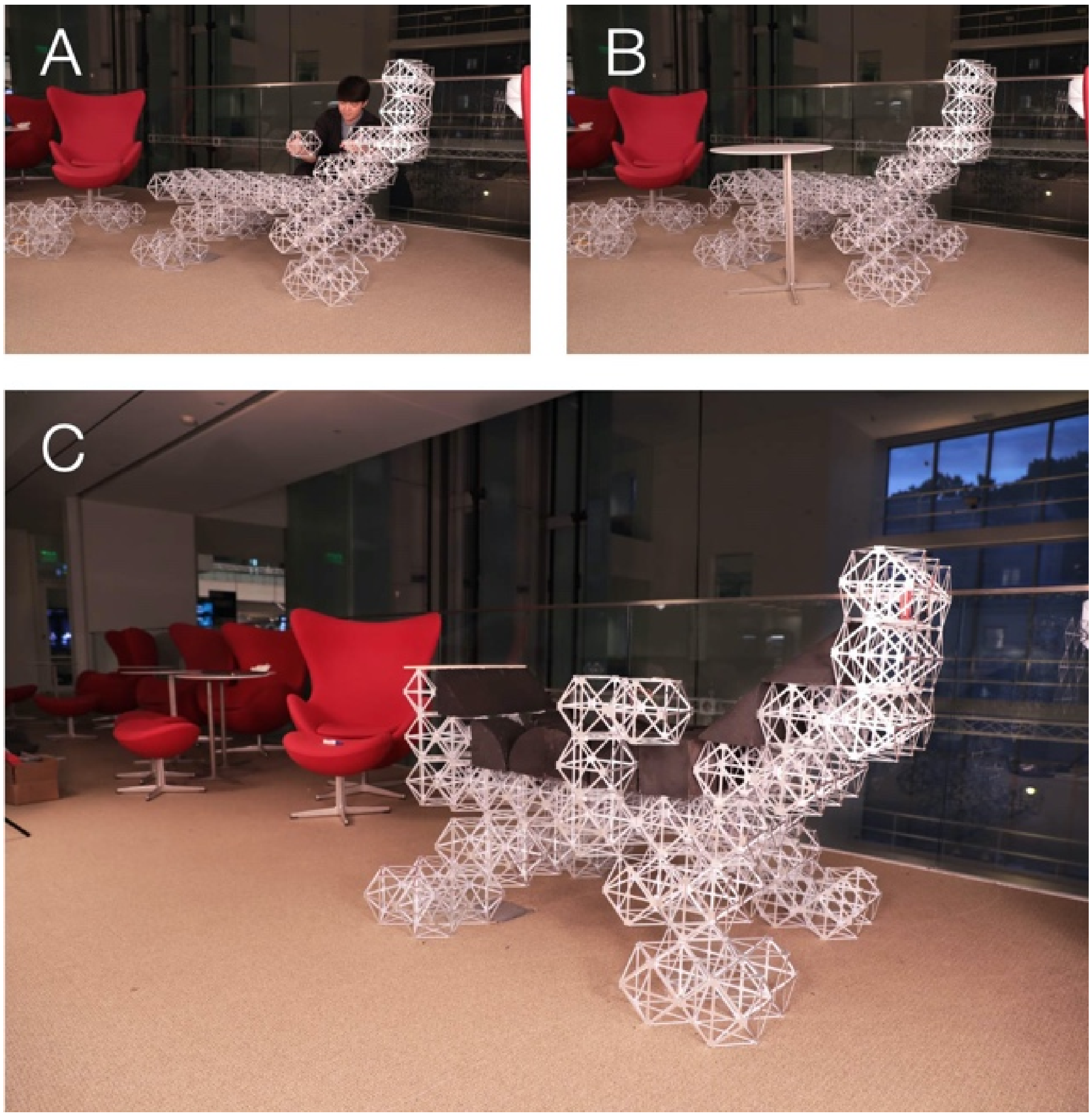

Figure 2-18: A) Assembling the basic frame of the lounge chair. B) Testing chair height and comfort in situ against other furniture. C) Iterating based on that to add a table for a computer and armrest before outfitting the structure with soft foam for comfort.

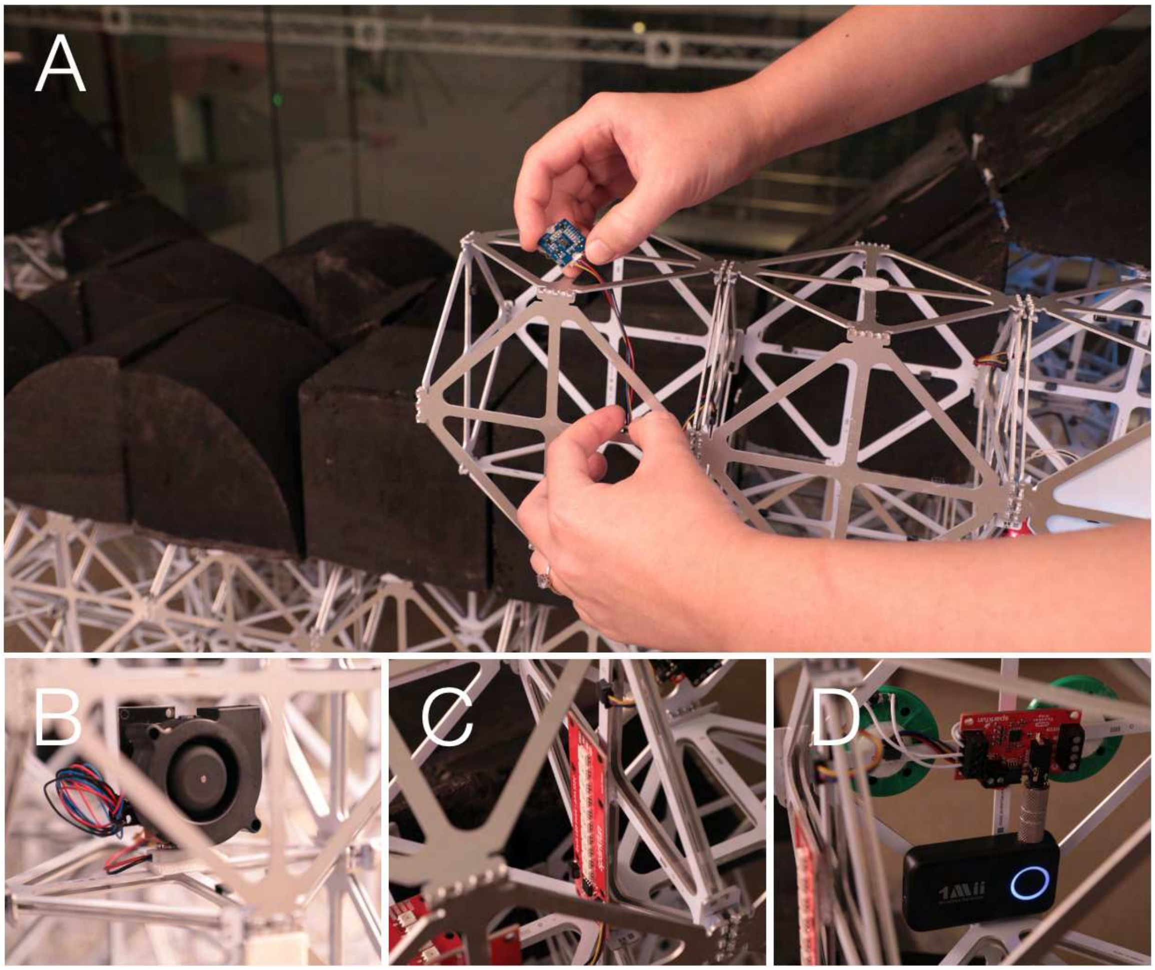

Figure 2-19: A) Installing a Grove Gesture sensor via the Qwiic connectors. B) A Qwiic-controlled PC fan for ventilation, with its own dedicated battery. C) Networked LED strips for a controllable reading light. D) A Qwiic and Bluetooth-enabled speaker module for hands-free audio control.

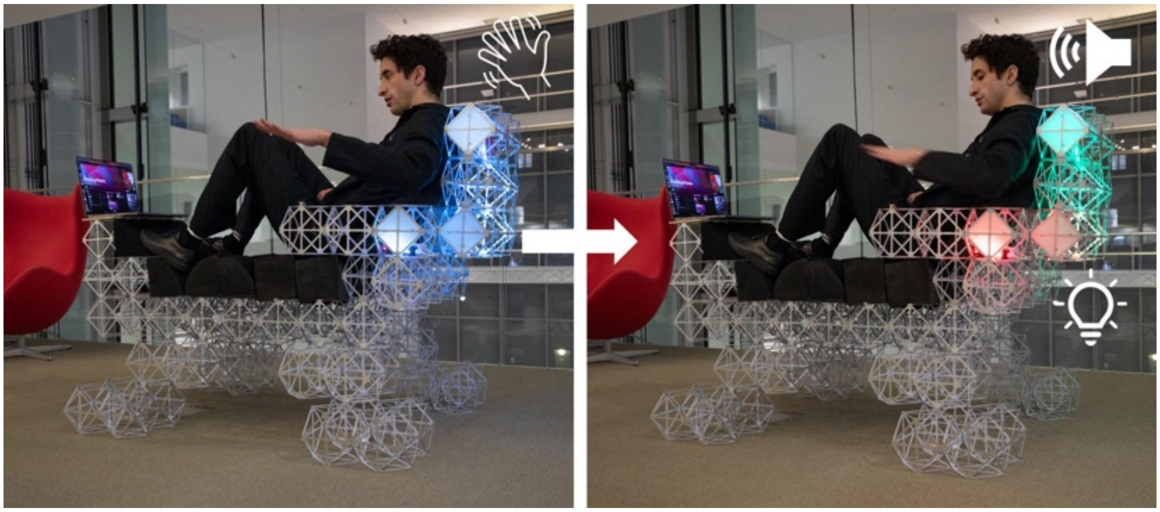

Figure 2-20: Inputting gestures to change light and audio settings.

a step toward broadening access to human-scale electronics-integrated prototyping.

2.8 Discussion, Limitations, and Future Work

2.8.1 Sensing and Actuating on the Lattice

The lattice structure offers potential opportunities for direct sensing integration, such as adding strain gauges or capacitive sensing into either the voxel frame of the clip system, especially with the exploration of other voxel geometries, as in [91]. Future work can open new dynamic interactive modes by adding actuators such as servos or liquid crystal elastomers [62]. Integration of such actuators would also extend the system's usability for modular robotics systems, similar to [17], or toward self-assembly systems as in [161]. Actuation that takes advantage of the lattice system could also be explored, such as in [132].

2.8.2 Optimizing Mechanical Performance and Scalability

The current mechanical performance of VIK is sufficient for it to be of interest for larger-scale applications beyond what we explore in this work. For example, given the durability of aluminum as a building material, VIK could provide an interesting basis for outdoor electronics prototyping, integrating solar panels for power. In the future, we will work on improving aspects of the system so that it is usable for commercial or industrial applications beyond a prototyping tool. Specifically, we can expand the types of networking supported by VIK beyond an I2C bus, which, while convenient for prototyping, has limited functionality at larger scales. In practice, we find that a single I2C bus on VIK supports total distances of approximately 1 m before path resistance and capacitance become limiting, which could be addressed by switching to a different network, such as CAN bus. For structures with hundreds or more devices on them, it may be beneficial to switch to a communications strategy that doesn't require addresses, such as hop-count, similar to what NeoPixels implement, or a self-discoverable network such as in [149]. Additionally, the joints and faces may be further optimized via both topology optimization approaches and user-study-informed approaches to push the performance of the system even further.

2.8.3 Increasing Accessibility

One of the goals of this work is to open the use of high-performance mechanical systems to people without engineering backgrounds or access to engineering tools. By designing the system to use aluminum PCBs, users can access VIK cheaply and quickly via commoditized PCB manufacturing processes. For this work, we produced approximately ~250 voxels, which is already a high unit count, enabled by the ease of this manufacturing approach.

Our current approach still requires users to be comfortable with Arduino-level microcontroller programming, which leaves out many people. Future work will address improving the usability of this system for electronics novices, such as by leveraging [149], a hardware virtualization system to enable easy microcontroller networking for machines.

2.9 Conclusion

We introduced VIK, a kit for creating large-scale, structural, reconfigurable 3D electronics. Our system uses aluminum PCBs, produced by an accessible commercial process, that assemble into lightweight, high-stiffness voxels to create a myriad of interactive devices. Because of these features, VIK has the potential to eventually scale to much larger systems than what we have so far demonstrated. We developed a design tool to help users create and simulate VIK structures, reducing the engineering expertise a user needs to deploy a high-performance mechanical system with integrated electronics quickly. The reconfigurability of the system enables users to take apart a structure and reuse the constituent parts if they no longer need the original structure without sacrificing mechanical performance. Combining the electrical and mechanical elements minimizes the amount of material that goes into VIK constructions. Though even without electronic functionality, VIK still stands as a robust and lightweight building block in its own right as a fully reusable/reconfigurable building system.

Chapter 3

Robotic Voxels for Robotic Voxel Assembly

The work in this chapter appears in part in the publication "Self-Reconfigurable Robots for Collaborative Discrete Lattice Assembly", which was written with my collaborator Amira Abdel-Rahman and published in the ICRA conference proceedings from 2024.

3.1 Abstract

We present a robotic system for the assembly of 3D discrete lattice structures in which the robots are able to self-reproduce, such that the assembly system may scale its own parallelization. Robots and structures are made from a set of compatible building blocks, or voxels, which can be assembled and reassembled into more complex structures. Robotic modules are made by combining actuators with a functional voxel, which routes electrical power and signals. Robotic modules then assemble into reconfigurable robots via a reversible solder joint. The robot assembles higher performance structures using a set of construction voxels, which do not contain electrical features. This paper describes the design, development, and evaluation of this assembly system, including the robotic hardware, lattice material, and planning and controls methods. We demonstrate the system through a set of fundamental assembly tasks: the robot assembling another robot, and the two robots collaborating to assemble a small structure.

3.2 Introduction

The autonomous assembly of large, high performance structures remains limited by the size and complexity of the necessary machines. As the size of the machine gets larger, the demands on the relative precision of the machine also get larger; the machine must travel larger distances while maintaining a low error threshold in material placement. Assembly-based approaches to building large structures using discrete elements can address this scaling problem by introducing error-correcting features into the discrete feedstock— the precision of the assembled material corrects for imprecision in the assembler [46]. Mobile assembly robots can be used to decouple assembler size from the size of the final structure, enabling the construction of arbitrarily sized structures [89]. This style of robotic assembly has been demonstrated with both highly complex robotic systems, such as with aerial drones [125] or arms mounted on moving bases [76], as well as with relatively simple robots, by co-designing the assembled material with the robot such that the underlying material corrects for errors in the robotic system [89]. This style of "relative robot" assembly often uses a robot which directly locomotes over the assembled structure, such as in [89], [142], [112], or which uses a combination of robots to deliver and place new material, such as in [121] or [78].

However, it is inefficient to build large structures if a robot must continuously traverse the entire structure to pick and place single new material units. To improve the efficiency, we can increase the parallelization of the assembly system by using a modular robot, enabling a robot to assemble either more of the target structure, more robots, or larger robots that are able to manipulate more material at once. In this way, the swarm can scale itself to the construction task at hand. In simulation, [17] demonstrates that taking this self-replicating hierarchical approach can optimize the efficiency of constructing large structures.

The goal of this work is to provide a functional hardware basis for realizing this style of robotic assembly system. We introduce a hardware basis for realizing the self-reproducing robotic assembly system described in [17], addressing physical limitations of the robots used in [89] and [17] that prevented them from assembling load bearing systems or self-replicating, respectively. In our system, as in [17], the assembly robot itself is composed of modules compatible with the material it assembles (see Fig. 3-1), and we have iterated both these modules and the building material system (or voxels) so that they form structural connections. We demonstrate this system through the assembly of small-scale structural voxel primitives and through the robotic assembly of a second robot that can then assist in further voxel assembly.

3.3 Background

Cellular materials or architected materials represent a solution for distributing material efficiently and effectively in space. In engineering applications, these materials are often used in systems that need to be lightweight without sacrificing mechanical performance. Cellular materials span a range of sizes and purposes— at the very large scale, we might recognize a truss bridge as a cellular material assembly, and at the other end of the size scale are micro-lattices with features on the order of 100s of microns [157]. Much of the current research in producing this type of structure is focused on materials with very small features— lattice pitches on the size scales of 100s to 1000s of microns, which then form material samples on the size scale of centimeters [156] [157] [184]. The manufacturing techniques required for producing material at this scale, such as two-photon lithography

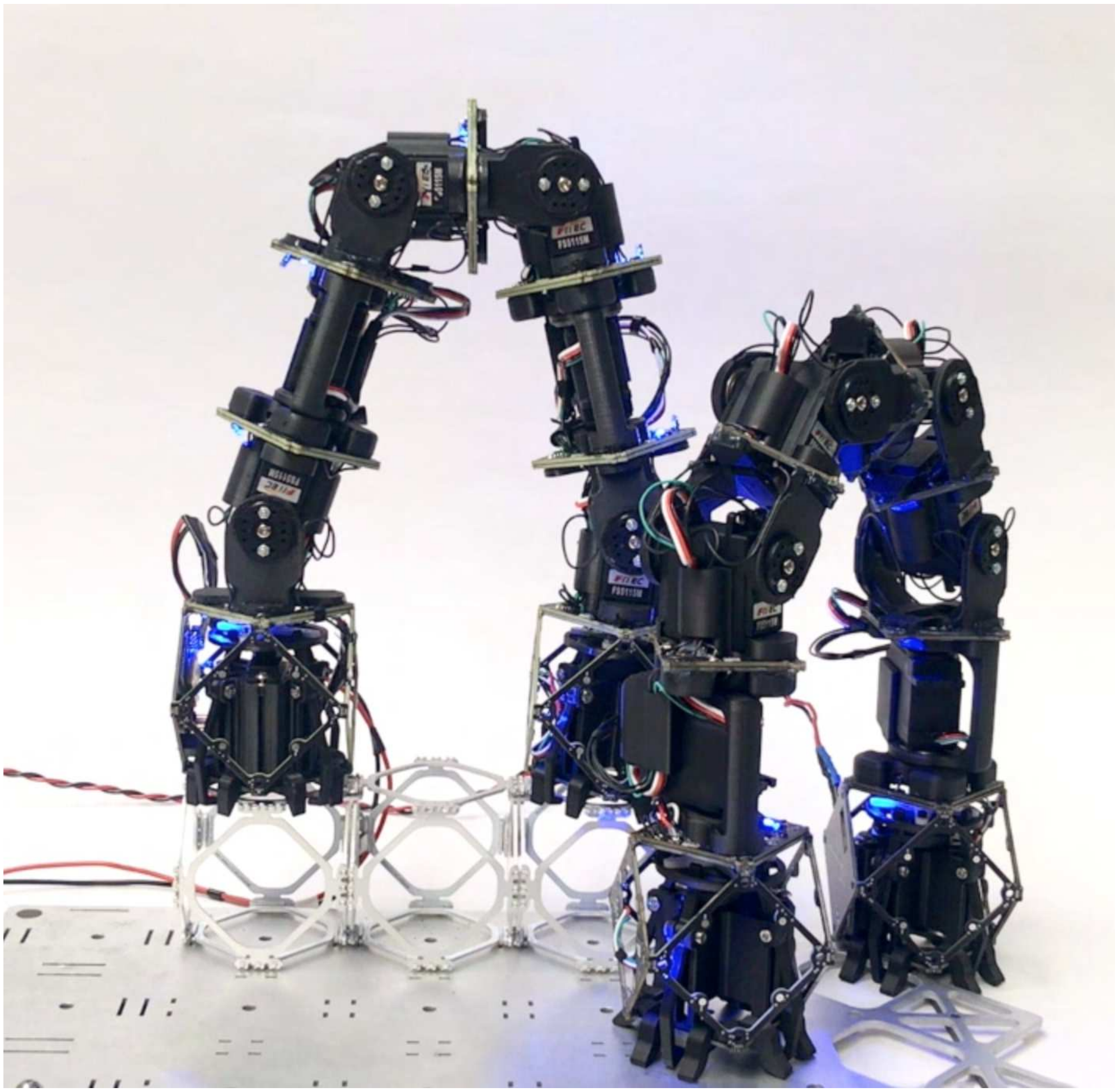

Figure 3-1: Two robotic assemblers standing on a build plate that provides a stable starting point for the assembly of larger voxel structures. The robot on the right was assembled by the robot on the left.

3D printing, would be difficult to scale up to sizes past the 10s of centimeter scale [91]. Discretizing the lattice into building blocks which can then be assembled into arbitrarily large structures can leverage the benefits of cellular materials at larger size scales [46].

This work focuses on methods for automating the assembly of cellular materials at large scales using a voxelized approach. In prior work that demonstrated discrete cellular material for mid- to large- scale assemblies, such as for soft robotic [138], automotive [96], or aerospace [90] applications, the assemblies were composed manually— limiting the utility of these systems for broader application. Introducing a robotic assembly system would enable the adoption of the materials for wider use. To maintain global accuracy over the voxel structure, we adopted a relative robot approach to assembly [89], in which our robot and material systems are designed together, such that the material system is able to correct errors in the positioning of the robotic system. A prior robotic voxel assembly system, BILL-E [89], [17], [93] used a magnetic voxel feedstock with self-correcting features on the robot grippers to achieve consistent part placement in spite of the simplicity and minimal feedback present in the system. Similarly, brick- based systems, such as TERMES [142], a termite-inspired swarm construction system, or VaultBot [121], have also used specially designed bricks, both with magnetic insets, which interface with the assembly robot in specific ways to improve assembly performance.

Much of the prior work in modular robotics has also been concerned with developing modules that are able to be reconfigured in repeatable ways by an imprecise assembler— the modular robot itself. Robot modules typically have well-defined attachment points, leading toward chain- like geometries, in which modules attach end to end such as in [178], or grid-like geometries, in which modules can attach on multiple faces, such as in [167], though some work has also explored module attachment at arbitrary locations, such as [168]. These systems have employed a variety of attachment processes, such as motor-driven latching, as in [106] or [33], or phase change systems, such as through reflowing solder in [129], or melting a polymer [168]. The modular version of BILL- E [17] that this work extends on used a magnetic joint for the robot modules, which resulted in a low-strength connection that prevented the self-reproduction of that system and thus recursive assembly. Similarly, since the prior system only worked with magnetically connected voxel feedstock, the resulting structures could not bear load. The primary contribution of this paper is to demonstrate in hardware a modular robotic assembly system that is able to both assemble structures with desirable mechanical performance, and to assemble further robots.

3.4 Robot Hardware

3.4.1 Functional Voxels

The basis of the robotic system is a functional voxel— a building block element containing electronic functionality. [17] and [42] use a functional voxel made from PCB laminated to acetal, which uses soldered finger joints for the internal face connections and a combination of spring connectors and magnets for the voxel-to-voxel connections. For greater strength in that system, the magnets may be swapped for e.g. rivets, as done in [42], which renders the system more difficult to assemble robotically or to reconfigure. This work extends on the previously introduced design in [17] to incorporate a more mechanically stable joint compatible with robotic assembly, an increase in the amount of signals routed, and, optionally, the direct incorporation of microcontrollers into the voxel frame.

The functional voxel used here consists of six FR4 PCB faces assembled into a cuboctahedron or "cuboct" geometry. Similar to the work in [17], the joints within a voxel are epoxied and soldered. To enable autonomous connections between voxels without introducing significant additional weight, the system uses a reversible solder joint, modeled off of the system used in SolderCubes [129]. Like that system, the voxel faces have exposed pads primed with a low melt solder, and are heated using large surface mount resistors on the opposite side of the PCB. Unlike [129], we only use resistors on one side of the connection to reduce the amount of necessary electronic components, though it results in a longer attachment time— approximately six minutes to reliably flow and then cool the solder joint.

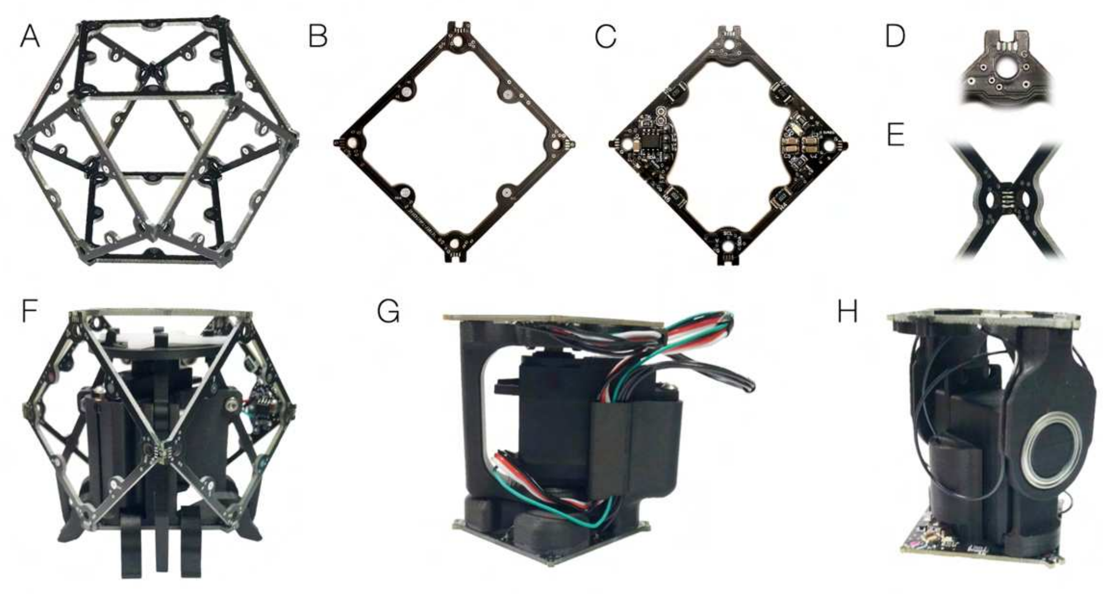

Figure 3-2: Functional voxels and robot modules. A) A fully assembled functional voxel comprised of six PCB faces. B) The basic frame face, which routes four lines and carries no components. C) The ATtiny412 face, which controls the motor in each module. D) A close up view of the inner-voxel solder joint of one face. E) A close up view of a soldered inner-voxel joint. F) The gripper module. G) The wrist module. H) The elbow module.

This system uses three main types of functional voxel faces. The first is a basic frame, which routes four signals— ground, supply voltage, serial clock, and serial data— and carries no additional electronic components. We then use two types of microcontroller faces: a primary controller face and a generic robot module face. The primary controller of the robot uses an

ESP32-WROOM microcontroller, receiving commands over Wi-Fi from a central computer and disseminating these to the microcontrollers on each robot module over an I2C bus. The robot modules use ATtiny412 microcontrollers, which receive commands over the data lines and can either move a servo, report a measured joint angle, or activate the resistive heating circuit.

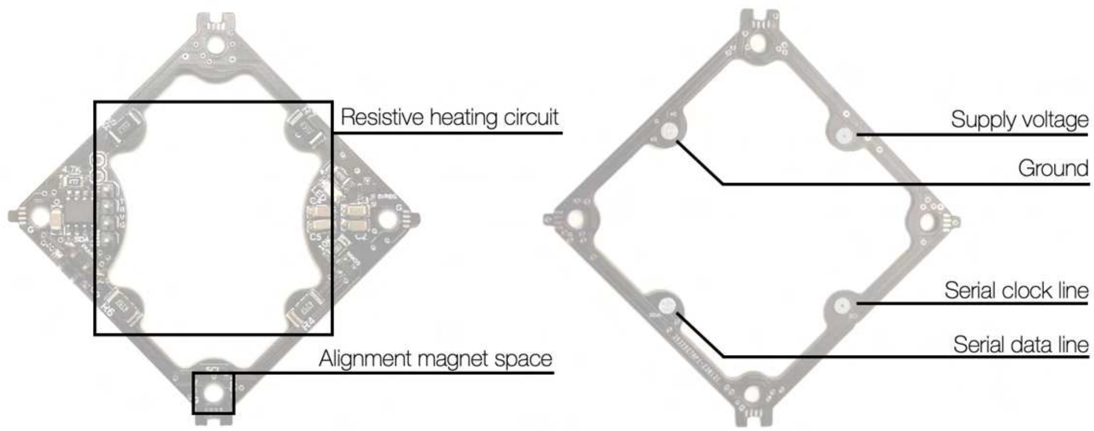

Figure 3-3: Labeled components of two of the functional voxel frame types. (Left) ATtiny412 face with resistive heating circuit and alignment magnet install locations demarcated. (Right) Basic frame with the reflow solder signal pads labeled.

3.4.2 Modular Robot

Using this set of faces, we can build robotic modules by adding actuation. The robotic assembler makes use of three types of modules: a gripper, a wrist-type joint, and an elbow-type joint. The gripper attaches to voxels, enabling the robot to locomote over the lattice or to manipulate voxels. The wrist and elbow modules are both rotational joints with approximately 180 ° range of motion. To select an appropriate motor for the modules, we considered the potential worst case loading scenario— in which the robot cantilevers its entire length while carrying an additional module— and calculated the approximate required motor mass and stall torque required. Based on this, the FEETECH High-Torque Servo FS5115M-FB from Pololu was selected as a low cost and convenient option for the wrists and elbows with the feedback line broken out. The grippers use the smaller FEETECH Mini Servo FT1117M-FB from Pololu as the torque requirements are lower. The modules use FDM 3D printed parts to interface between the functional voxel faces and the motors. Parts were printed in either PLA (for the configuration 1 style robot, as shown in Fig. 3-4) or nylon with chopped carbon fiber (for the configuration 2 robot).

Modules are assembled into robots by stacking. The wrist and elbow type modules only accept connections from opposing ends (as in a chain-type modular robot), while the grippers can be configured to accept connections from any of the faces aside from the gripping surface. Though the robots explored in this project only make one connection from the grippers, future versions of the robot could add additional features by expanding from the grippers. Specifically, a separate gripping arm (as in [17]) could be added to improve the payload carrying ability of the robot. This paper explores two configurations of 6DoF robot, both of which use two grippers, two wrists, and four elbows. These configurations are shown in Fig. 3-4— in the first, there is an elbow joint immediately above the grippers, and then a wrist joint, while in the second the wrist joint comes first. The configuration 1 robot has a longer vertical reach than the configuration 2 robot, while the configuration 2 robot has a wider set of diagonal movement available to it.

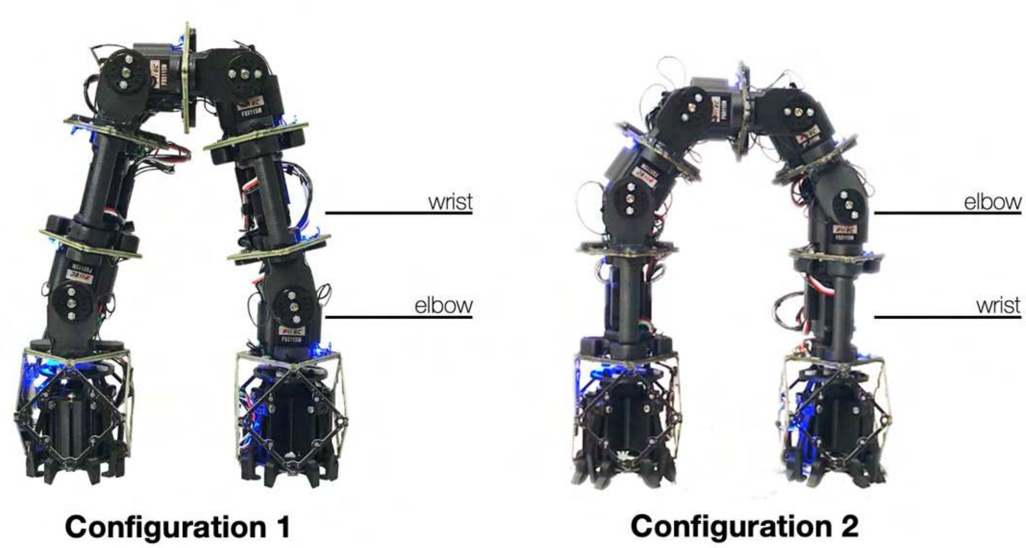

Figure 3-4: Two configurations of robot that we explored. The different configurations have different accessible spaces, and so are better suited for different types of movement from each other.

3.5 Construction Voxels

We used a face connected cuboctahedron unit cell, as done in [91]. The unit cell is decomposed into six 2D faces which are snap-fit assembled together and then decorated with latching features to facilitate voxel-to-voxel connections. Voxels are designed to be installed and removed vertically, for ease of robotic assembly. Because the voxel design is based on 2D faces, it is compatible with a range of materials. For this work, we primarily used aluminum voxels or acetal voxels. The aluminum voxels were made either from laser cut aluminum or were purchased as aluminum PCBs from JLCPCB. PCB manufacturing was chosen as it is a cheap process— individual faces were priced at less than 0.30USD per face at 250 order quantity. Acetal voxels were laser cut.

Two clip families were used. In the first, the clips constrain all but one degree of freedom

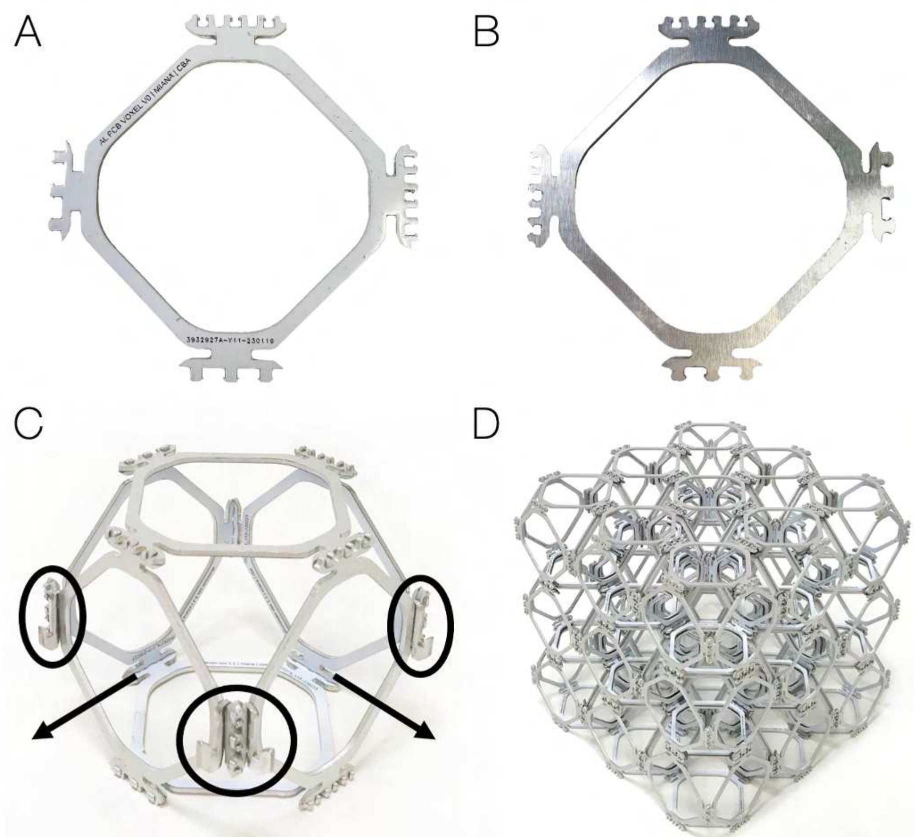

Figure 3-5: The construction voxel system. A) A single aluminum PCB voxel face from the top view. B) The bottom view of the same voxel face. C) A voxel with the lateral connection clips highlighted and the assembly direction demarcated. D) A 3x3x3 assembly of voxels for characterizing the behavior of the bulk material.

once a voxel is placed, and require the installation of an additional clip to fully constrain the voxel. In the second, the clips are designed with snap fit features, such that once placed, the voxel is fully constrained. A variety of vertical clip types were explored, though all utilized snap fit features to attach to the lower voxel.

Figure 3-6: Some clips explored for this system. (Clockwise, from top left corner) A vertical connector clip, another variant of the vertical connector clip, a snap fit lateral joint, a smaller snap fit lateral joint, a shear clip, a small press-fit joint, a larger press-fit joint, another variant of vertical clip, and yet another variant of the vertical clip.

The voxels were mechanically characterized through Instron testing. We primarily used acetal voxels for the testing to characterize the behavior of the joint and voxel geometry. For this, we tested 1- and 2- voxel stacks under tension and compression, with fixtures matching how loads would be applied to a voxel in a lattice. We tested the joints under tension, in shear, and under cyclic loading. We additionally tested the aluminum voxels under compression to examine the bulk material behavior and scaling. Some of the key results from this testing are summarized in Table 1. Specifically, the modulus (MPa) and load (N) are reported for 1- and 2- acetal voxel assemblies tested in compression and tension (each of these tests is repeated three times), as well as for acetal and aluminum 2x2x2 and 3x3x3 voxel assemblies (which are only tested once). For complete testing results, see [165].

The immediate goal of the construction voxels is to provide a lightweight load bearing system— which the current design succeeds at— with a secondary goal of achieving behavior equivalent to a continuously manufactured cuboctahedron lattice— which the current design falls short of. The 3x3x3 aluminum structure has a density of 62.5 Kg/m^3, a compressive modulus of 5.1 MPa, and reached a maximum load of 4576.68 N, placing the lattice's mechanical behavior in a similar range to some metal lattices in [95], or the plastic and composite lattices in [184] or [91].

However, for this lattice geometry, we would expect better performance, as well as a larger improvement in the modulus as the voxel count increases, as reported for the systems described in [72] or [91]. To reach this ideal behavior, the behavior of the voxel structure needs to be dominated by the behavior of the beams, as opposed to the joints. Because of the current design of the clips and interfacing features in the voxels, this is not achieved.

Table 3.1: Summarized Voxel Mechanical Properties

Part	Material	Test	Modulus (MPa)	Max. load (N)
1x1x1	acetal	C	0.404 ± 0.009	137.29 ± 1.45
1x1x1	acetal	T	0.367 ± 0.012	210.56 ± 22.22
2x1x1	acetal	C	0.372 ± 0.003	134.52 ± 0.46
2x1x1	acetal	T	0.314 ± 0.009	82.55 ± 26.52
2x2x2	acetal	C	0.518	513.30
3x3x3	acetal	C	0.522	1055.14
1x1x1	aluminum	C	4.50	353.27
2x2x2	aluminum	C	5.07	2028.4
3x3x3	aluminum	C	5.10	4576.68

A single acetal voxel in tension reaches a maximum load of 210.56 ± 22.22 N, while two acetal voxels in tension (which now have clips between them), only reach a maximum load of 82.55 ± 26.52 N before failure, clearly indicating that the clips are a weak point in the system. Similarly, the variance in the maximum loads reached in compressive tests is much lower than that of the tensile tests, indicating that the clips behave less consistently than the voxels. The indexing features on the voxel faces, as well as the clips themselves, introduce additional compliance in the system that prevents optimal behavior.

3.6 Controls and Simulation

We developed a custom web-based design and simulation tool to be able to test different control and path planning strategies for the assemblers, as well as further explore different robot designs and architectures. Fig. 3-7 shows an interactive interface that does the inverse kinematics for the robot given a target end position.

We implement the centralized path planning algorithm and the shape compiler introduced in [89]. The algorithm calculates the steps needed to build an arbitrary geometry given one or multiple voxel deposit locations. These steps are later sent to the physical robots to start assembling the structure. Using this algorithm, the robot can build any connected geometry that does not have underhangs, layer by layer. It is possible for horizontal layers to have unconnected areas, as long as all voxels are connected to voxels in the layer beneath them. We also implemented the spatio-temporal scheduling scheme introduced in [89] to do path planning for multiple robots. Fig. 3-8 shows an example of two robots collaborating to build a spanning structure.

Fig. 3-9 shows some robot design explorations using the simulation tool. Using the set of modules that we introduced, one can design robots that are able to carry a larger number of voxels, or collaborate to carry out harder tasks that cannot be achieved by a single robot.

The custom design tool makes it easier to explore and simulate these variations in an easy parametric manner.

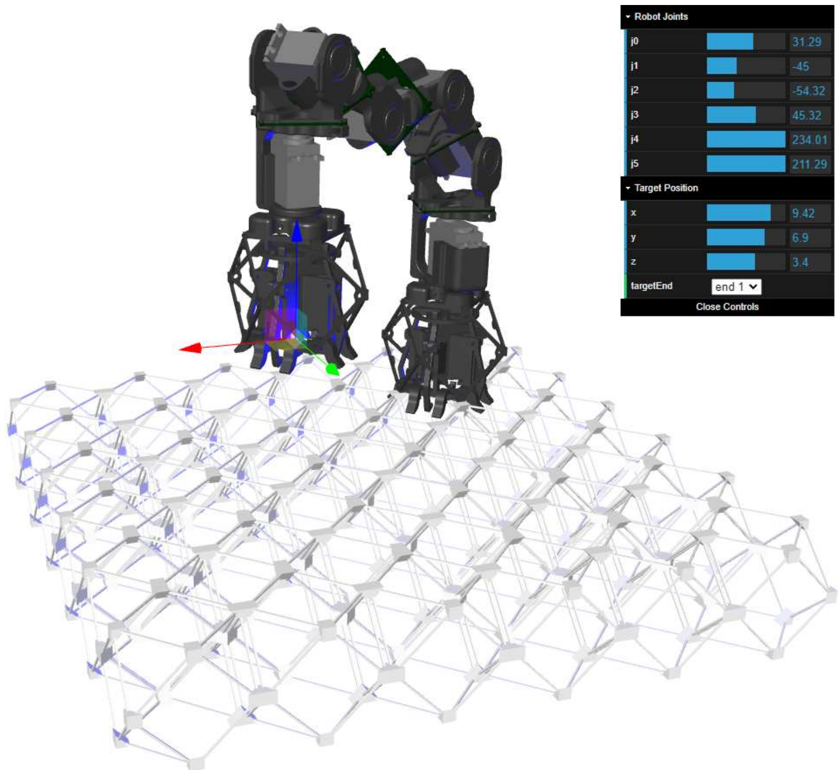

Figure 3-7: Screenshot of the inverse kinematics web interface. The robot end effector position may be changed by dragging the axis, or by toggling the joint position.

3.7 Robotic Assembly System

3.7.1 Assembly demonstrations

The robotic system was primarily evaluated through demonstrations of fundamental assembly tasks— the construction of a 1D beam, a 2D cube, and another robot. We used the robots to build beams and small cubes (2x2x2 voxels) out of aluminum voxels. Fig. 3-10 shows the basic steps for voxel assembly. The robot starts standing on the build plate and a new voxel is manually fed in. The robot swings around to pick up the new voxel, grabs it, and places it. If using the snap-fit clips, then this placement is stable, otherwise, an additional clip is required. A new voxel is then fed in.

This basic sequence is also used for assembling additional robots. New robot modules are fed into the manual feed location. After the placement of the first module, which must contain the second robot's ESP32 primary controller, power is wired to the second module. The

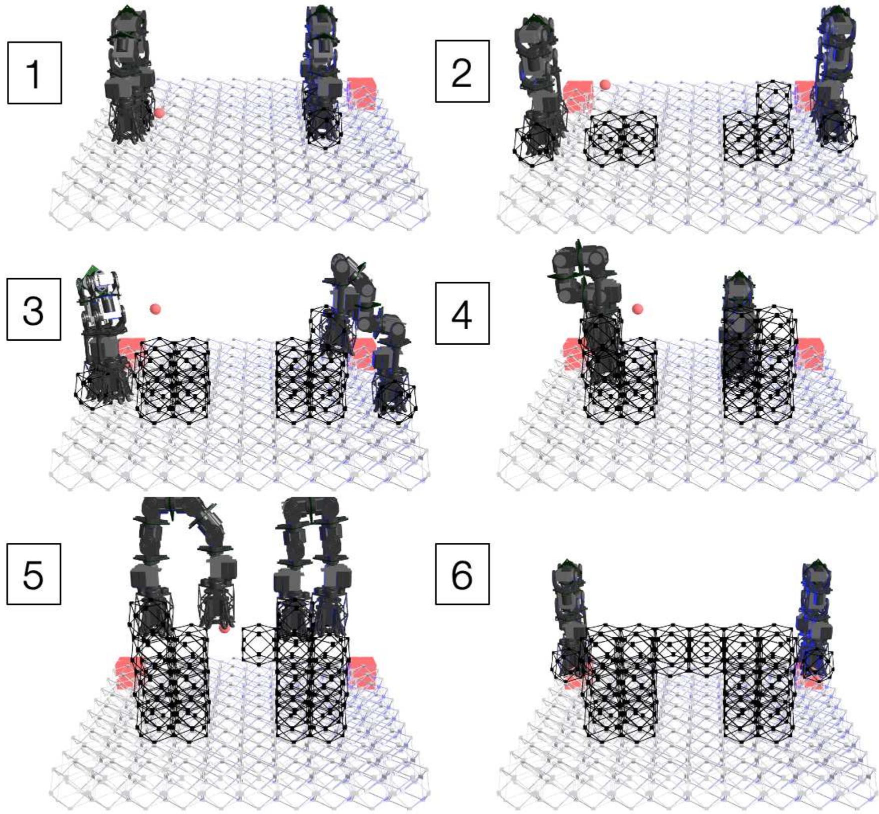

Figure 3-8: Stills from the path planning tool of two robots assembling a spanning structure.

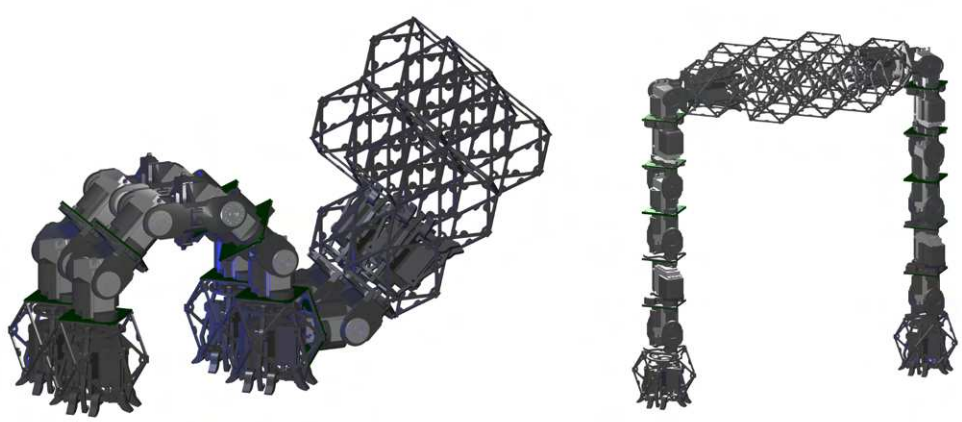

Figure 3-9: Renderings of notional alternate robot geometries with larger payload carrying abilities from the design tool. (Left) a parallelized version of the current assembler, shown carrying 8 voxels. (Right) Two of the current robot configurations holding 9 voxels together, acting temporarily as one larger robot.

first robot can then stack modules to build out the robot, with the second robot running its resistive heating routine after the placement of each new voxel. Magnets installed in the faces of the modules, as well as some extending 3D features in the modules, help with consistent module alignment, in spite of imprecision from the assembling robot.

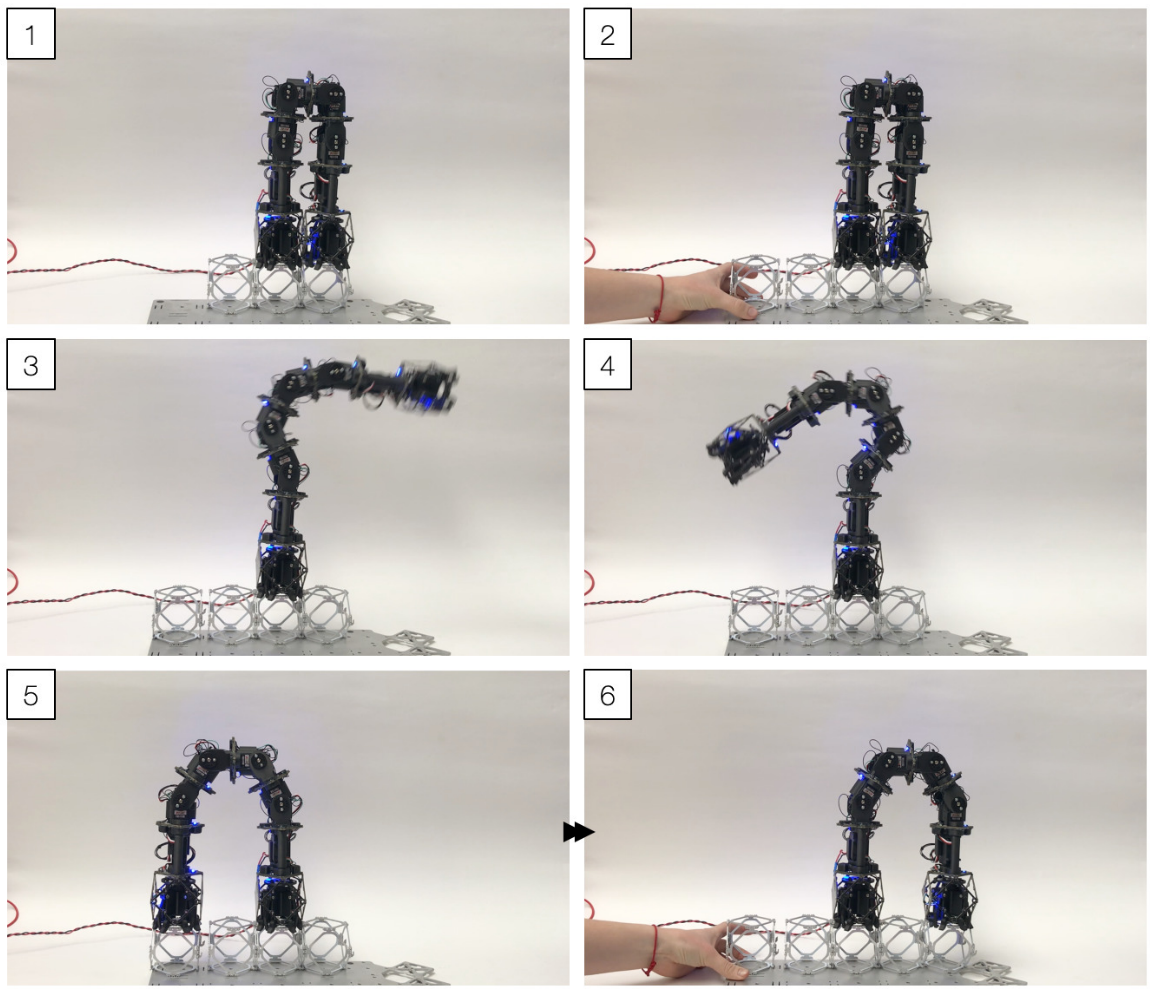

Figure 3-10: The basic assembly sequence. 1) The robot starts standing on the build plate. 2) A voxel is manually fed in. 3-5) The robot swings back to grab the voxel. 6) The robot places the voxel and a new voxel is fed in.

At a certain point, the assembling robot is no longer able to reach the next module placement for the robot it is assembling. At this point, the assembled robot can help the assembling robot by moving to accessible positions, as shown in Fig. 3-11. Fig. 3-11 shows this assembly sequence for the attachment of the last two modules. The robot that is being assembled moves itself so that the assembling robot can reach the next placement location, and then moves out of the way, so that the assembling robot can pick up the next module. This is continued until the second robot is complete.

Once there are two robots, the robots may collaborate with each other to improve the

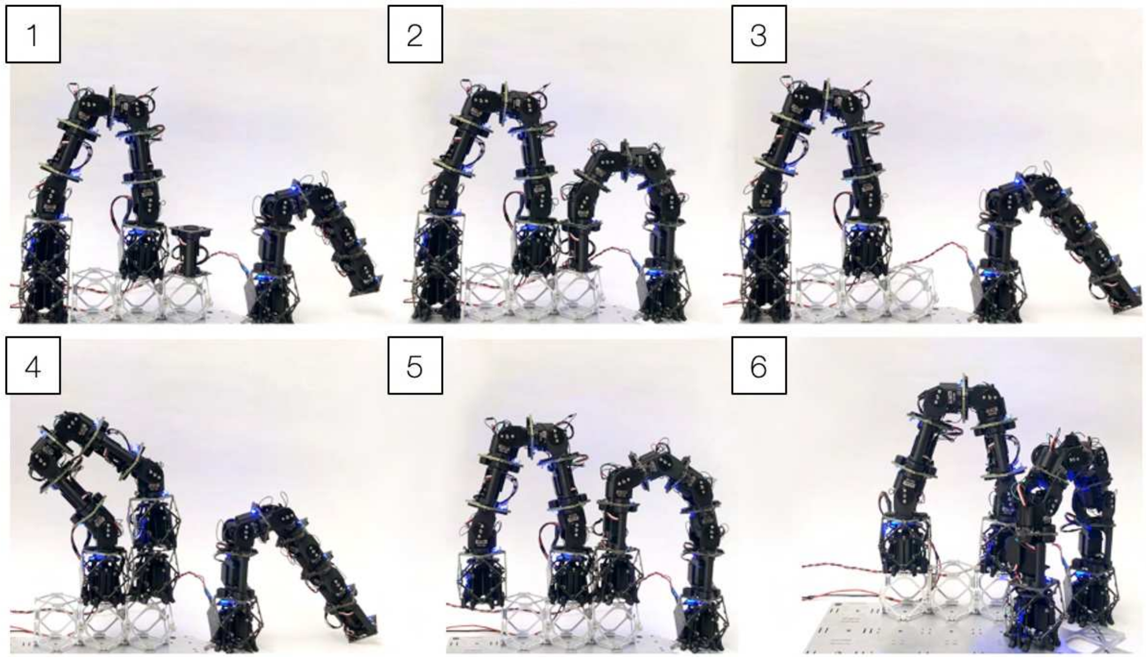

Figure 3-11: A portion of the robotic assembly of a second robot. 1) The 7th module of the assembled robot is placed on the lattice. 2-3) the assembled robot connects to it, and 4) the assembler places the last module. 5-6) The last module is connected to form two robots.

overall performance of the system. We demonstrate a basic version of this collaboration by having both robots work on the assembly of a four voxel cantilever beam that supports the weight of the assembly robot (see Fig. 3-12 for reference). The robots initially build out a 3-voxel beam from the build platform, as done in Fig. 3-10, and then they add another 2 voxels on top of that beam, allowing the second robot to step up onto the initial cantilever. From there, the first robot picks up and passes voxels to the second one, which then places those voxels to extend the cantilever.

As is visible in Fig. 3-12, the beam does visibly deflect under the weight of the robot, which would negatively impact the ability of the system to continue accurately building out the beam for a different geometry (e.g., if the beam were built to bridge between two voxel structures, the deflection could potentially prevent the accurate indexing of the voxels onto the other side). Currently, the path-planning systems do not account for this type of non-ideal behavior. Future work will address this by iterating the material system to make it stiffer, as well as potentially advancing the path planning software to account for local material deformations during assembly.

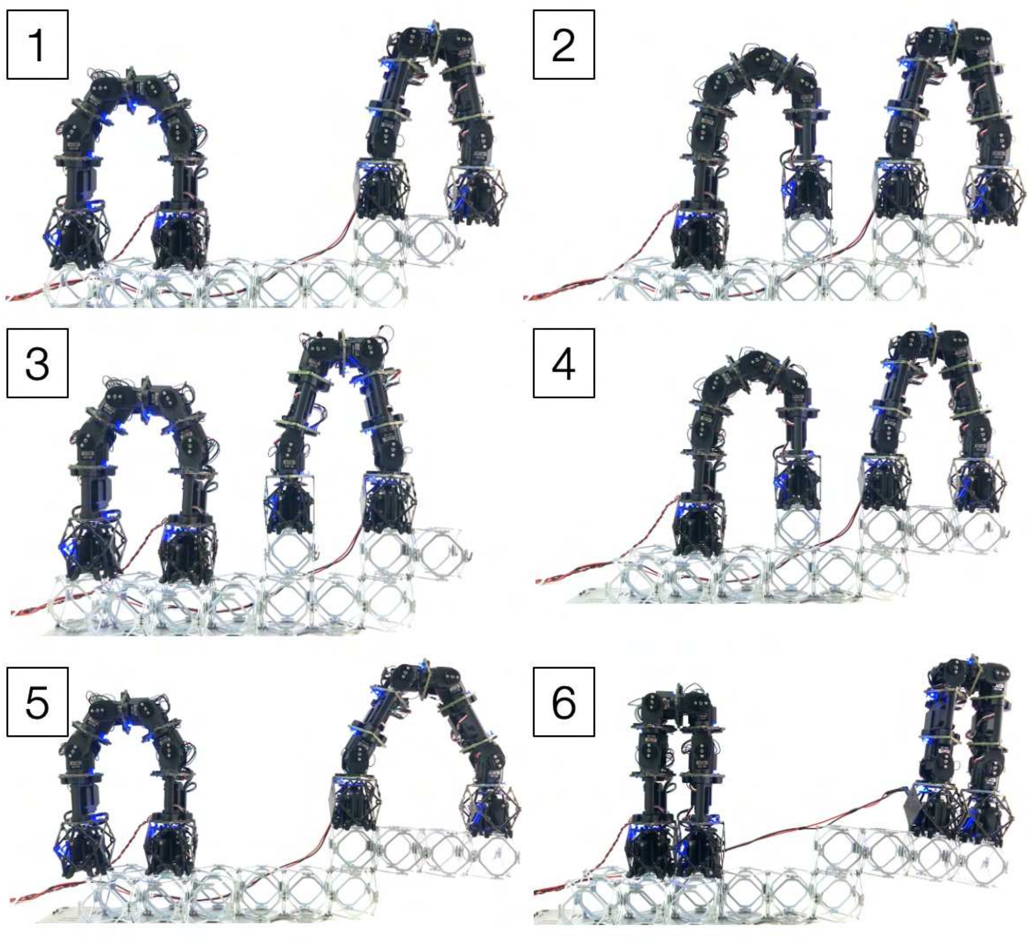

Figure 3-12: Collaborative robotic assembly of a cantilever beam that supports the weight of one robot. 1) Robot 1 picks up a voxel and 2) passes it to robot 2. 3) Robot 1 returns to the voxel feed location, while robot 2 grabs the voxel left by robot 1. 4) While robot 1 delivers the next voxel to robot 2, robot 2 extends the cantilever. 5) Robot 1 reverts to its initial position and robot 2 builds out the cantilever. 6) Robot 2 walks to the end of the cantilever beam.

3.8 Discussion

The robotic assembly system demonstrates in hardware the basic functionality for a self-reproducing assembly swarm: the robot can make another robot, and these robots can build load bearing structures, advancing it past the prior BILL-E assemblers this work extends on. This work is a starting point for this system, though, and so still faces challenges before it can fully realize the goal of an autonomously self-scaling swarm construction system.

The relative robot building strategy assumes that the interface between the robot and the material is reliable enough to correct any failures in the positioning of the robot. To evaluate this assumption, the repeatability of this system was tested by repeatedly walking the robot back and forth on a voxel structure. Specifically, the configuration 2 style robot walked back and forth over a voxel beam, using three different step types. The overall failure rate was 7.1 %. These failures occured primarily because the robot could not detect if a step failed—further iteration of the gripping leg flexure design would help mitigate failures, while the addition of a minimal feedback system, such as the switch used in [93], could allow the robot to self-correct.

Because the robot does not have additional voxel carrying capabilities, to build beyond its reach, it must place voxels in temporary locations, and then pick and place them to their final location. This is less efficient than simply carrying the voxels, and introduces more steps during which the robot could fail. We will likely add a third gripper to the robot, as done in [9] to resolve this.

We only demonstrate collaborative robotic behavior in a limited manner. Even though for a structure with this few voxels, the use of two robots does not substantially improve the time efficiency of the system, this represents a small-scale demonstration of the type of collaborative robotic behavior we want to further develop, such that it enables the efficient assembly of eventually large structures. The current system does not use online planning, that is, path planning is done on a central computer and exported to the robots that then execute, analogous to running g-codes on CNC machines. Future work could explore online planning algorithms, such as those described in [50], and work up towards replicating in hardware the results described in [17], specifically in exploring hierarchical robot configurations.

Chapter 4

Hierarchical Discrete Lattice Assembly

The work and text in this chapter appear in part in the publication "Hierarchical Discrete Lattice Assembly: An Approach for the Digital Fabrication of Scalable Macroscale Structures" which is written with my co-authors Paul Richard and Alexander Htet Kyaw. I would like to specifically note that the voxelization and planning tool was developed by Paul, but I include it in this chapter for a holistic description of the voxel assembly system. This was published at the ACM Symposium on Computation and Fabrication in 2025.

4.1 Abstract

Although digital fabrication processes at the desktop scale have become proficient and prolific, systems aimed at producing larger-scale structures are still typically complex, expensive, and unreliable. In this work, we present an approach for the fabrication of scalable macroscale structures using simple robots and interlocking lattice building blocks. A target structure is first voxelized so that it can be populated with an architected lattice. These voxels are then grouped into larger interconnected blocks, which are produced using standard digital fabrication processes, leveraging their capability to produce highly complex geometries at a small scale. These blocks, on the size scale of tens of centimeters, are then fed to mobile relative robots that are able to traverse over the structure and place new blocks to form structures on the meter scale. To facilitate the assembly of large structures, we introduce a live digital twin simulation tool for controlling and coordinating assembly robots that enables both global planning for a target structure and live user design, interaction, or intervention. To improve assembly throughput, we introduce a new modular assembly robot, designed for hierarchical voxel handling. We validate this system by demonstrating the voxelization, hierarchical blocking, path planning, and robotic fabrication of a set of meter-scale objects.

4.2 Introduction

Machines for digital fabrication typically cannot make objects or structures larger than themselves. Although this is not necessarily limiting on the desktop scale, as target structures scale toward human or architectural scales, the prospect of ever-growing machine footprints eventually becomes untenable [119]. Current digital fabrication approaches for larger-scale structures often section the target structure into parts that can be produced separately and then assembled, usually manually [63] [105], [31]. Approaches that consider assembly automation often rely on large and expensive industrial robotic arms with limited overall mobility [124]. In either case, there is no clear strategy toward scaling to unlimited build footprints or autonomous environments.

In contrast, collective robotic construction, in which a group of robots collaborate to assemble a structure larger than any individual robot, offers significant promise towards scale-agnostic, efficient, and autonomous fabrication [144]. However, hardware demonstration of these systems has typically been limited in scale [89], load capacity [143], or geometric freedom [121]. This is because automated large-scale fabrication with mobile robots is inherently challenging, with many competing priorities, such as structural stability versus mass efficiency, geometric complexity versus ease of robotic fabrication, or robotic functionality versus swarm simplicity.

By designing a material system together with its robotic assembly system, we can address some of these issues. The mechanical interplay between the assembly robots and the underlying material system can enable local error correction, allowing relatively simple robots to build large and precise structures [89]. In this work, we present an approach to the assembly of large structures using small relative robots that manipulate hierarchical blocks of lattice material. Our approach uses a new type of discrete lattice block and robotic system aimed at improving assembly throughput and mechanical stability. We use standard digital fabrication processes to pre-produce compounded blocks of architected lattices– a high efficiency material system– at the size scale of tens of centimeters, and then use mobile robots to take over beyond that, to the meter scale. To ensure structural stability, we developed a tool similar to a 3D printing slicer that imports a 3D mesh, voxelizes it, and finds interleaved patterns for robots to assemble. It then simulates the assembly live, streaming instructions to the hardware system. Our contributions are:

- An interactive tool for voxelizing 3D shapes and finding connected patterns for different shapes of blocks.
- A digital twin path planning and simulation environment integrated with the hardware system, for both assembly control and live design and re-design of structures.
- The development of a compounded set of self-aligning, interlocking, and load bearing octet lattice blocks designed for robotic assembly.
- The development of a new modular inchworm style assembly robot, designed for hierarchical assembly and low system cost and complexity.

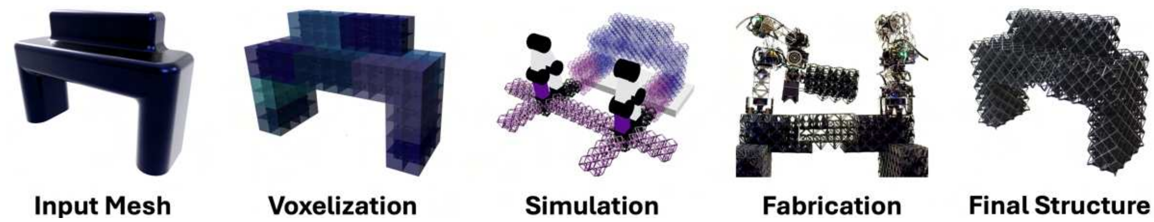

Figure 4-1: Pipeline from STL Mesh to voxelization, to simulation, to robotic assembly, to the final structure for a small bench.

4.3 Related Work

Our approach builds on existing research in scalable building systems, robotic construction and assembly systems, and material-robot and modular robotic systems. In this section, we overview these fields and identify trends and limitations that guided our approach in this work.

4.3.1 Scalable Building Systems

Although this work is focused on the robotic assembly of structures, scalable prototyping systems designed for human use offer useful insights into reliable mechanical design for building at larger scales. Specifically, we draw from systems that are not significantly limited by the build volume of the fabrication systems used. Broadly, these systems tend to fall into three categories: continuous, sectioned, and discrete fabrication processes.

Continuously Fabricated Systems

An example of a continuous fabrication system capable of producing meter-scale structures is Protopiper [21], which extrudes tape-based pipes to draw out human- to architecture-scale structures. However, this system has limited load bearing capacity. Wire benders, such as [140], are similarly able to extrude much larger structures than the machine itself, and with potentially good load-bearing capacity, such as in [35]. However, these continuous extrusion based processes have no mechanisms for preventing error accumulation in the built structure, and instead rely on the precision of the machine, or human intervention, to correct problems as they arise— a potentially significant issue, given that a small error in bend angle over a long distance can result in substantial deflection from the target shape.

Sectioned Fabrication and Assembly

Splitting up a target structure into subcomponents can help ameliorate some of the issues with error accumulation, while additionally opening up a wider array of fabrication systems.

These approaches are typically geared toward breaking a larger model down into either 2D components that can be cut and assembled together, or smaller 3D models that can be separately fabricated and assembled. LuBan3D [116] is a software package aimed at doing exactly this: facilitating users to fabricate structures much bigger than the machines they have, such as the house in [155]. More specialized examples aimed at non-expert users include Kyub [31], which enables the design and automatic creation of load-bearing structures made from laser cut plywood, or HingCore [19] and PopCore [18], which offer joinery-free methods for processing foamcore or equivalent sandwich panel material. The primary issue with these methods is that they produce highly specific parts for specific designs, often requiring dexterous methods for assembly. This makes automation much harder: handling extremely diverse geometries in many degrees of freedom is a hard problem in robotics. Additionally, this typically prevents re-use of parts, as components are likely not portable to new designs, requiring full re-fabrication and re-assembly to make design iterations.

Discrete Assembly

Continuing the trend toward breaking up a big structure into smaller parts is the class of systems that use discrete assembly, where a single part type (or a small library of component types) is used to iteratively build many different target structures. A classic example of this is LEGO®, which enables an imprecise assembler (e.g. a child) to build precise and relatively robust structures. This is because the material system is designed to self-align and error correct during the assembly process, so that the accuracy of the final structure is primarily determined by the tolerances in the part manufacturing, and not by the tolerances of the assembler, in contrast to continuous extrusion fabrication processes. LEGO® uses a volumetric brick-based decomposition, but other forms exist, including strut-and-node systems [105], facet-based assemblies [95], and voxel-based assemblies [72]. This style of system lends itself well to robotic fabrication, as it features a limited and similar set of parts, often designed to accommodate imprecise assemblers.

Among discrete assembly approaches for large-scale structures, we focus on using discrete architected lattices, as these systems offer competitive mechanical properties at a lightweight [156]. This is desirable both to improve the performance of the resultant structure and to lower the payload demands on an assembly robot, which reduces some engineering challenges associated with heavy building materials [69]. Discrete versions of architected lattices have demonstrated extreme performance results [46], designed material anisotropies [91], and integrated electronic functionality [162]. Manually, these have been assembled into a wide array of meter-scale structures, such as morphing aero- and hydro- structures [90] [138], vehicles [96], or load-bearing static structures [162], demonstrating the broad applicability that is desirable for a generalist building system.

4.3.2 Robotic Fabrication Systems

We now consider the question of how to automate the building of large structures. Approaches for this, again, roughly fall into three categories: large machines to build large structures, large robots to assemble large structures, or small mobile robots to assemble large structures.

Static Gantries

In the first approach, a typically gantry-based machine is used to assemble a structure, as in [26], or, more commonly, to 3D print a structure, as shown in [30] often using systems in the style of [49], for example. Though these approaches have been demonstrated at commercial architectural scales, they are still fundamentally limited by the need for very large machines, which introduces both an upper limit on the size of the structure that can be made, as well as significant logistics challenges, either in terms of transporting parts, in the case of pre-fabrication, or building the machine in place, in the case of in-situ fabrication. With the extrusion based approaches, there are additional open challenges in material formulation and performance [118] [150]. For these reasons, we are interested in further exploring assembly based approaches.

Industrial Arms

At the large, complex, and expensive side of the assembly based approaches are projects that have focused on using industrial robotic arms for assembly. These have demonstrated a range of material types, such as steel rebar [117], timber [27], or bricks [68], including commercial systems such as [13] or [11]. However, these systems have limited accessible footprints, based on the reach of the robotic arm (or the linear axis it is mounted on), or feature significant localization challenges, if on a mobile base [38]. These projects often rely on the high performance of their robotic and sensing systems to compensate for the simplicity in their building materials— by shifting some of the complexity onto the material system, we can reduce some of the demands (and size and cost) on the robotic system.

Material-Robot Systems

Building systems that rely on the co-design and interaction of the robots and material systems are often referred to as material-robot systems [89]. The degree of material-robot integration varies across the literature. At one extreme end, self-reconfiguring modular robots, as in [129] or [75], could be considered a material-robot system where there is no distinction between the building system and the robot, resulting in a very complex material system. At the other end, systems such as [69] or [112] only shift a minimal amount of complexity onto the material system.

For discrete lattices, systems for voxel-based assembly, such as in [89], [161], or [71], as well as strut-based assembly, such as in [78] or [179] have been explored. [89] demonstrated that an inchworm style robot could accurately traverse and assemble a voxel structure with no global feedback, based on the error correction between the voxels and the robot. However, this system relies on magnets and an underlying steel table to form connections and provide mechanical stability, and has low load bearing capacity on its own. This system is evolved in [137] to demonstrate the assembly of 100s of voxels [71]; however, the complexity of the robotic system is substantially increased, requiring three separate high degree of freedom robotic systems to perform assembly (a voxel carrying robot, a voxel installing robot, and a voxel fastening robot internal to the lattice), while the size scale of the lattice limits shape fidelity (300mm pitch).

In both of these voxel assembly systems, as the structure grows, the assembly throughput decreases, as the robot must traverse the built structure back and forth to pick up and install new material. [17] proposes that recursive hierarchical systems can help improve this issue— if assembly robots can assemble more assembly robots, then the swarm can increase its own parallelization, and if assembly robots can manipulate larger quantities of material at once, then they can build more efficiently. [161] demonstrates a first version of the hardware for a load-bearing modular voxel assembler, but is limited to desktop-scale objects.

Synthesis

Based on the prior art, we can distill some guidelines for an ideal robotic assembly-based digital fabrication system. Collective robotic approaches that have demonstrated the most promise towards scaling to practical applications typically strike a careful balance between the complexity of the material system and robotic system. The material system needs to be kept simple enough to be manufactured, while sophisticated enough to account for limited robotic functionality. The robot, in turn, needs enough features and degrees of freedom to reliably traverse and manipulate the lattice, but not so many that it becomes untenable to increase the amount of robots used. An ideal connection system only requires access from a single direction, without adding significant installation time. To maintain effective assembly times for larger structures, as well as to improve reliability, the system, as much as possible, should be able to parallelize its function and offer hierarchical assembly.

4.4 System Overview

We present a system for robotically assembling hierarchical discrete lattices. While prior voxel assembly systems have focused on a cuboctahedron lattice (equivalent to a node-connected octet lattice), we instead use an edge connected octet lattice because it inherently creates stable alignment features within each unit cell of the lattice (see Fig.4-2 for reference). By pre-connecting these unit cells laterally, we create compounded lattice/voxel blocks that can then tile to form stable 1D, 2D, or 3D structures (see Fig.4-3 for example tilings). These compounded blocks are thus all built from the same 1×1×1 unit voxel, but may vary in size and orientation (e.g., 2×2, 2×3, or 2×4), allowing a structure to mix different voxel types to optimize for geometry, performance, or assembly speed. The basic geometry of the lattice block is designed to self-align and constrain all but one degree of freedom when placed. In this work, the final degree of freedom is constrained with a releasable snap-fit (see Fig.4-6 for further detail).

The self-aligning compounded voxel block offers a few key advantages for robotic fabrication: 1) the alignment features permit placement error on the order of 1/2 the lattice pitch; 2) the installation of new voxels is purely vertical; and 3) the snap fit connection does not require physical access to the connection plane to engage, but is still reversible. Together, these features substantially reduce the requirements on the robotic system. To successfully build up a structure, the robots only need to traverse over the existing lattice, carry new voxels, and coarsely place them. To this end, we developed inchworm-style robotic arms that can crawl over the structure, extending on the paradigm used in e.g. [93], [137], or [17].

However, the use of compounded blocks introduces new challenges to the design and path planning of these structures. To this end, we developed a new pipeline for intaking standard 3D meshes, voxelizing them, grouping the voxels into compounded blocks, and then simulating their robotic assembly. The simulation space is integrated with the hardware, so that it additionally acts as the central control/coordination for assembly. This workflow is shown in Fig.4-1. The software system is further discussed in Section 4, while the hardware system is discussed in Section 5, with assembly examples and system evaluation in Section 6.

4.5 Software Implementation

A custom Web-based software environment was developed in JavaScript, enabling full control and simulation of the robotic assembly pipeline on any internet-connected device, without requiring local installation. Leveraging the Three.js visualization library, the system offers real-time 3D interaction through a synchronized digital twin of the robot. The pipeline begins with voxelization of a 3D mesh and proceeds through build sequencing, path planning, and real-time visualization with robot feedback.

4.5.1 Voxelization

The first step in the pipeline is the voxelization of the input geometry, any 3D mesh in STL format. The mesh is discretized using the size of the unit voxel, resulting in a resolution of 65mm. Once voxelized, we analyze the grid to identify repeating patterns composed of multiple adjacent voxels. These patterns can vary in size and orientation and may be

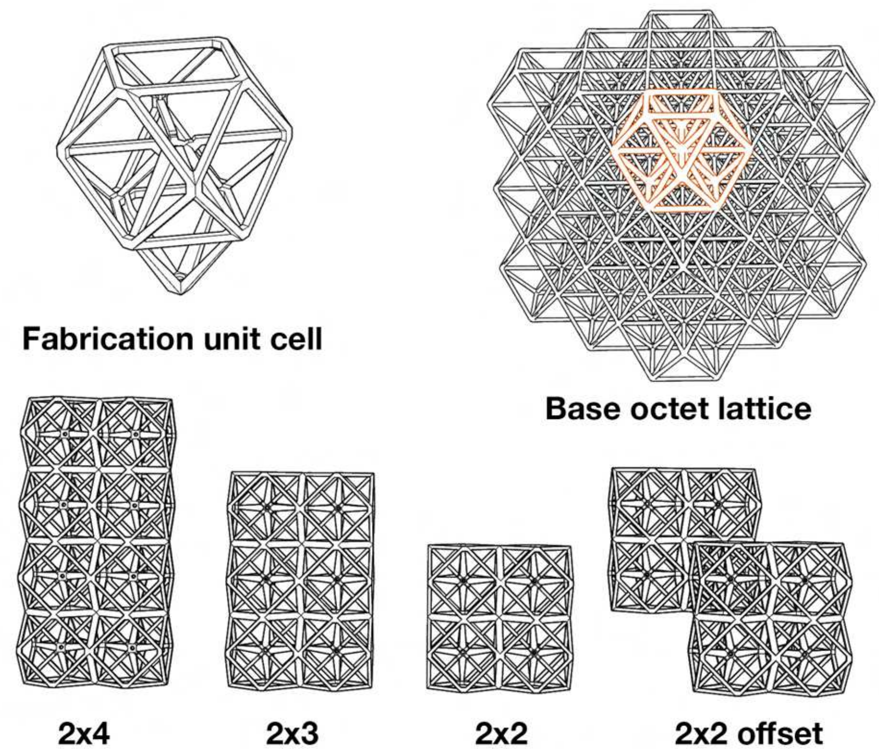

Figure 4-2: An overview of the lattice type and building blocks used in this project. The basic lattice type is an edge-connected octet lattice, which is decomposed into an extended cuboctahedron-octet, which is then compounded into different arrangements for robotic assembly.

composed of simple or stacked configurations. Some example patterns are detailed in Figure 4-2. To ensure connectivity and constructibility, the search prioritizes larger patterns first (e.g. 4x2x2 blocks) and proceeds hierarchically down to smaller units, filling in gaps as needed. Each detected pattern must be locally connected to at least one neighbor, either directly or via a base layer.

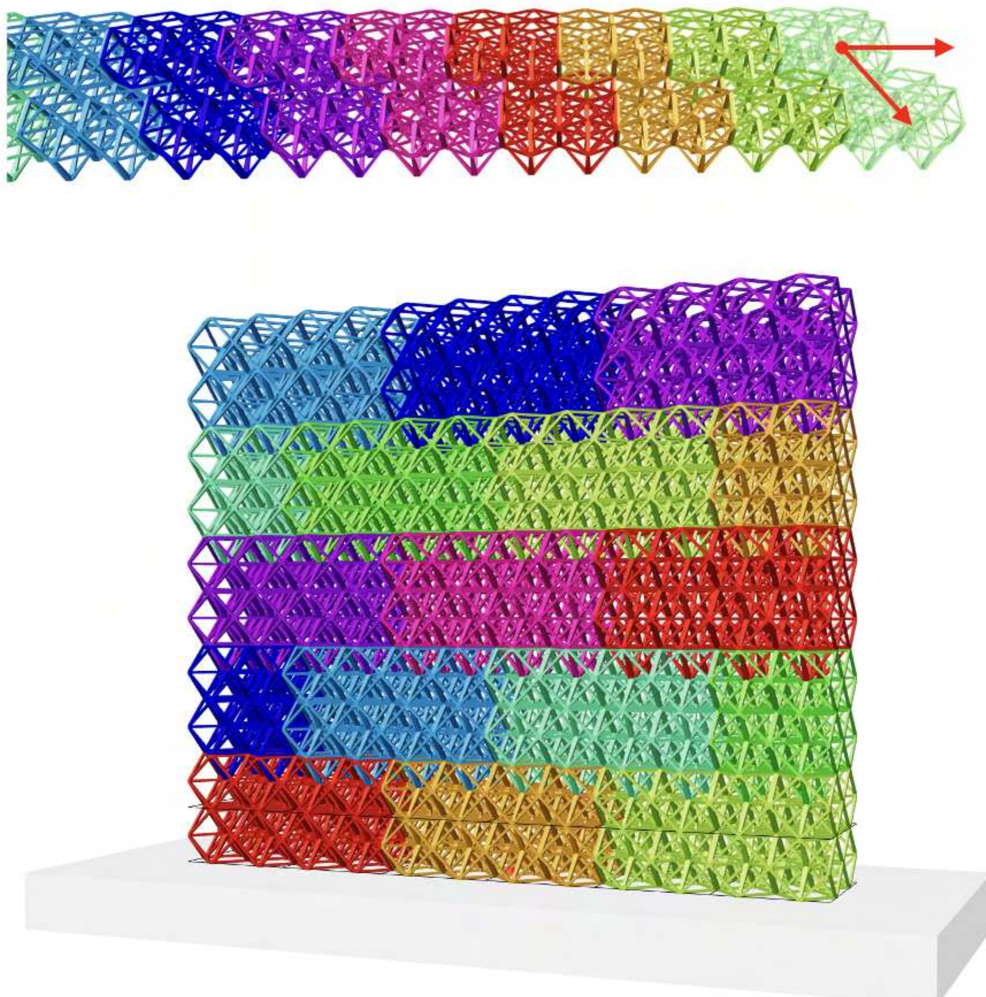

Figure 4-3: (Top) An example of tiling 2x2 offset voxel blocks to establish a first layer or overhang, with red arrows indicating the axes of potential extension. (Bottom) Beyond the first layer, layers can achieve interconnection through staggering layers.

4.5.2 Building Sequence

Given the location and type of each voxel in the structure, the next step is to define a build order that ensures feasibility while minimizing future path planning complexity. The building sequence determines the order in which each voxel of the structure is assembled and is influenced primarily by two parameters: the number of robots and the location of their associated voxel Feed. We assume each robot is assigned a unique voxel Feeder from which it collects and places voxels. The structure is first partitioned by assigning each voxel to the closest source based on the Manhattan distance. In case of equidistant voxels (a tie), a simple alternating policy ensures balanced distribution among Feeds.

Once assigned, the construction process is parallelized between robots. For each robot, a building sequence is computed independently. The structure is decomposed into horizontal layers, starting from the base and progressing upward. Within each layer, voxels are placed outward from the feed point, ensuring that the built region grows in a connected and stable manner.

This strategy guarantees that every newly placed voxel rests on an existing one from the layer below, thereby creating a new accessible surface on the current layer. As a result, the robot can step onto this newly created platform to continue placing subsequent voxels. The building sequence algorithm enforces this outward growth, ensuring that at every stage, the next voxel to be placed is both structurally supported and physically reachable by the robot.

4.5.3 Path Planning

Once the build sequence is established, each robot must autonomously navigate from its associated Feed location to its assigned placement position, taking into account the current state of the partially build structure. To generate feasible and collision-free paths, we employ an A* algorithm adapted to the discrete 3D voxel space, which minimizes the function:

$$h(n) = f(n) + g(n) \tag{4.1}$$

Where g(n) is the cost of moving from the start to voxel n and h(n) is a heuristic estimating the cost to reach the goal from n. We use the Manhattan distance as the heuristic, which performs efficiently in the voxel grid while preserving path optimality in our constrained setup.

The A* algorithm is particularly powerful in a voxelized environment, as the search space is inherently discrete, the connectivity is regular, and the heuristic directly reflects the grid geometry. By expanding nodes along the most promising directions first, it efficiently balances exploration and optimality. As a result, the algorithm consistently returns near-optimal paths with limited computation, even as the structure grows. Its effectiveness has already been validated in earlier iterations of this project [17, 161].

4.5.4 Robotic Assembly Simulation

Digital Twin for Live Simulation:

To monitor and validate the robotic assembly process in real time, we developed a digital twin environment that mirrors the simulated robot's actions on the physical hardware. Path planning outputs are executed in the simulation and simultaneously relayed to the robot via a Python middleware layer, using WebSocket communication over Wi-Fi. This architecture offers several key benefits:

1. Allows real-time updates during construction, enabling small-scale corrections in position or trajectory.

2. Provides live mapping of the robot's state and the evolving structure.

3. Offloads computation entirely to the simulation side, reducing the computational resources needed on the robot side.

This tool is used within the pipeline to align the simulation with the hardware using the data flow presented in 4.5.4, with all elements generated directly from the target structure. Nevertheless, it can also be used as a standalone application, allowing the user to manually place every element of the simulation (voxelized environment, robots, feeds, voxels to build) and subsequently link the configuration to the hardware. An example is presented in Figure 4-5.

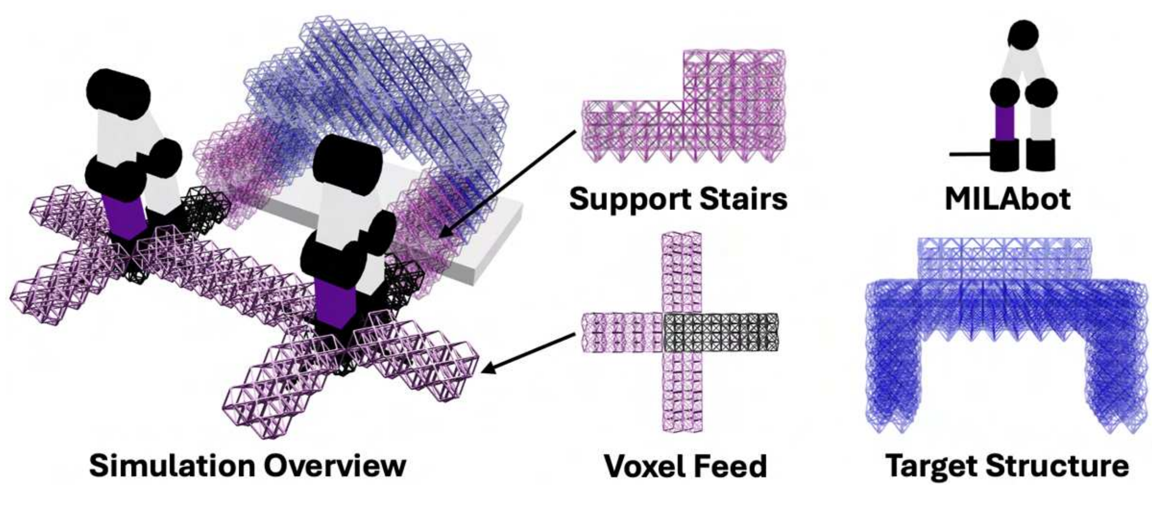

Figure 4-4: Simulation overview illustrating the four main elements: MILAbot, support stairs, voxel feed, and target structure.

Data Flow and Feedback:

The system operates using a feed-forward architecture: voxelization, build sequencing, and path planning are executed sequentially within the simulation environment. The resulting movement instructions are transmitted from the digital twin to the physcial robot— the Modular Inchworm Lattice Assembler robot (MILAbot)— via a Python bridge, which converts high-level actions into low-level joint target values using WebSocket communication. Simultaneously, the robot streams back its current joint states (see Figure 4-5). If a deviation from the expected pose is detected, a realignment sequence is triggered. This involves returning to the last valid position and re-approaching the target slowly to ensure accurate alignment without requiring full re-planning.

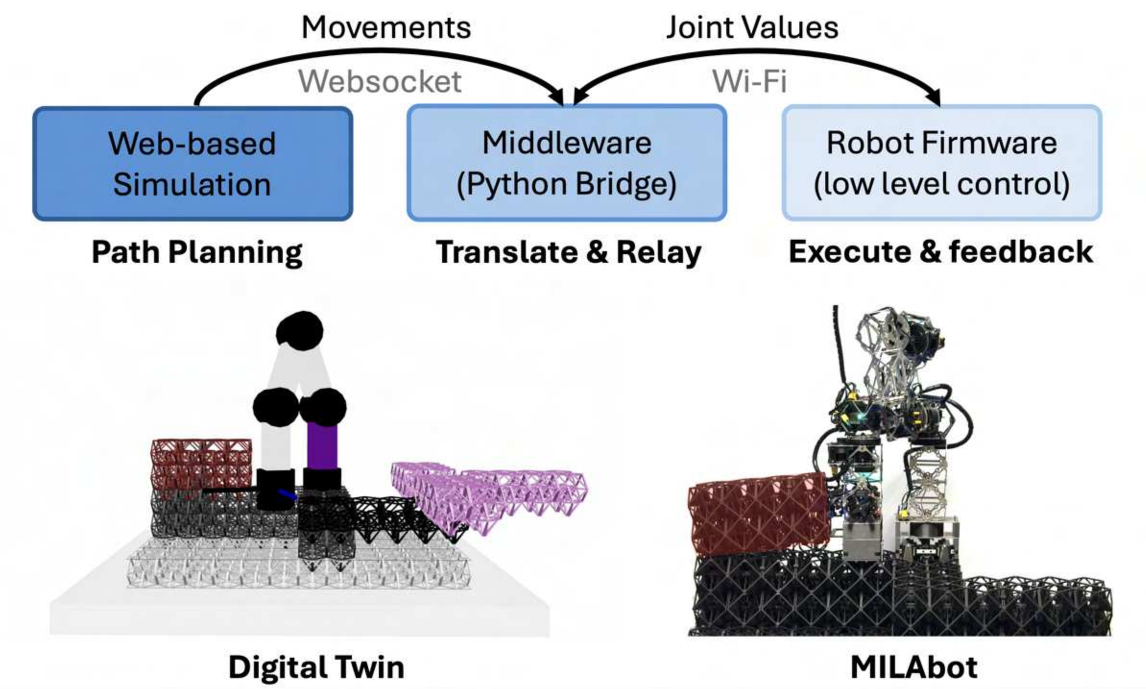

Figure 4-5: Top: Data flow from web-based simulation to MILAbot through middleware. Bottom: Digital twin synchronized with physical execution.

4.6 Hardware Implementation

In this section, we discuss the hardware implementation of the voxel assembly system, covering the voxel design, manufacturing, and performance, as well as the robotic system design and operation.

4.6.1 Material System

Our material system uses self-aligning compounded octet lattice blocks to create interlocking structures. In this work, we produce these lattice blocks in polylactic acid (PLA) through FFF 3D printing. We use FFF 3D printing because of its ability to easily handle complex geometries at a low cost and a reasonable speed. However, we envision that for future versions of this system, the compounded blocks might be further broken down into separate voxels, such as in [72], or individual faces, such as in [91], [161], or [95], so that they can be mass-manufactured from higher performance materials, such as GFRP in [91], CFRP in [71] or aluminum in [162].

To improve the printability of the voxels, we add flanges to support the large overhanging regions during printing, as done in [111]. The flanges enable clean print quality without the use of supports, enabling more efficient material usage as well as improved absolute mechanical properties of the voxel block. However, the flanges are not intrinsically necessary, and it possible to print the lattice in its unmodified form. We printed the compounded voxels in different overall sizes depending on printer availabilty and print bed-size, and if necessary, later laterally pre-assemble them with printed snap clips. A representative 4x2x2 compounded block is shown in Fig.4-6.

Voxel blocks are vertically connected using a screw-released snap fit connector. The snap fit consists of three parts: a set of pincers that by default sit too wide to engage with a voxel below it, a screw (in this case an M4 8 mm long socket head screw) to fasten it directly into the base of a voxel, and a screw hat, which acts as a spacer to push the legs of snap pincers out, so that they can engage with the voxels below (these are shown in Fig.4-6). The snap fit connectors are installed into the lowest octets of the printed voxels. The act of fastening the M4 screw then forces the pincers into the correct state, such that the voxel block can snap into one below it. To uninstall the connector, the M4 screw can be removed, which released the snap fit, allowing the block to removed. Though this is currently done manually, future versions of the robotic system could include an end effector for unscrewing the connectors.

So that the voxels sit on a flat surface, we additionally produced base-layer voxel blocks that consist of the upper half of the standard block. These are added onto the protruding octets to create a flat base layer without adding extra height to the system.

4.6.2 Assembly Robots

We introduce the Modular Inchworm Lattice Assembler robot, or MILAbot. As in prior voxel assembly robot systems, such as [161], [137], [89], the primary objective of the MILAbot design is to enable scalable voxel assembly with a system that can locomote over and assemble load-bearing discrete lattice structures. Looking toward scaling to the assembly of larger structures, the MILAbot is designed to increase assembly throughput while minimizing overall system cost and complexity. Additionally, the MILAbot is designed with

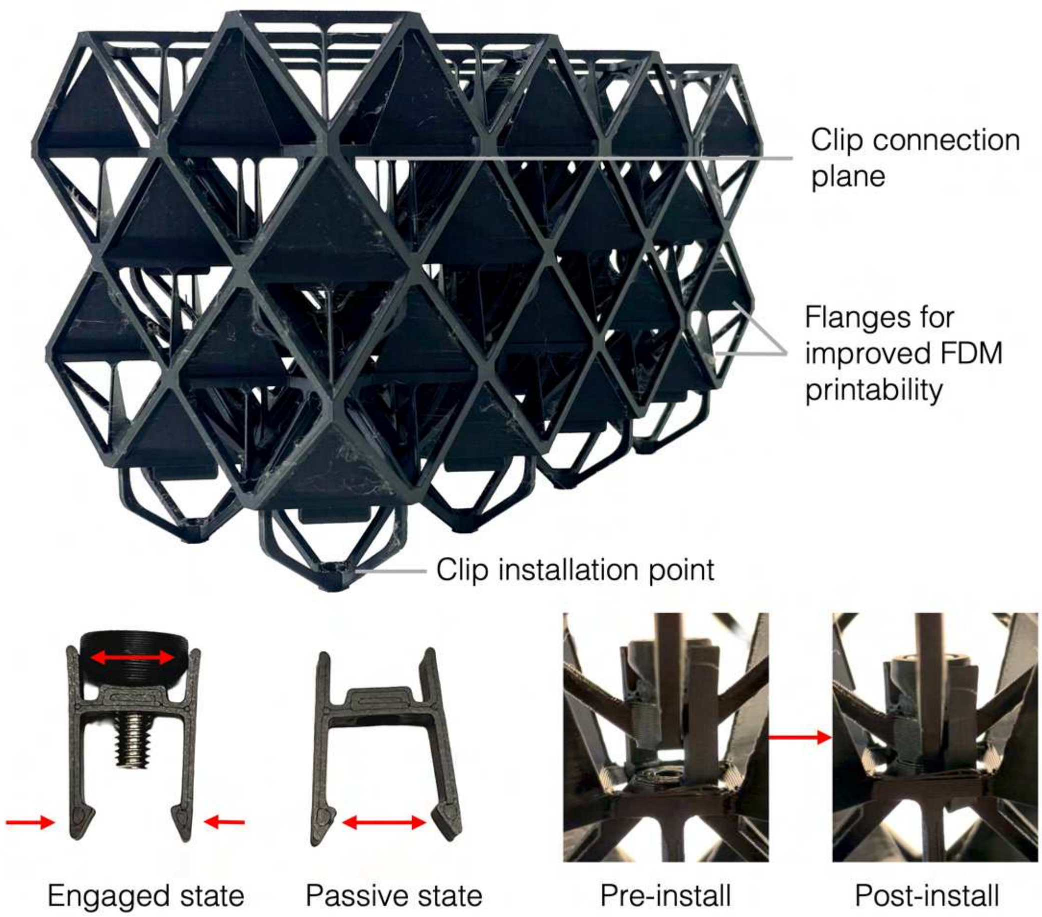

Figure 4-6: (Top) A 4x2x2 block of FFF printed PLA octet lattice with printability features added. (Bottom) Screw-release snap fit used for vertical connections.

eventual self-assembly in mind, that is, a starter MILAbot should be able to assemble more MILAbots, to autonomously improve efficiency as described in [17] and [165].

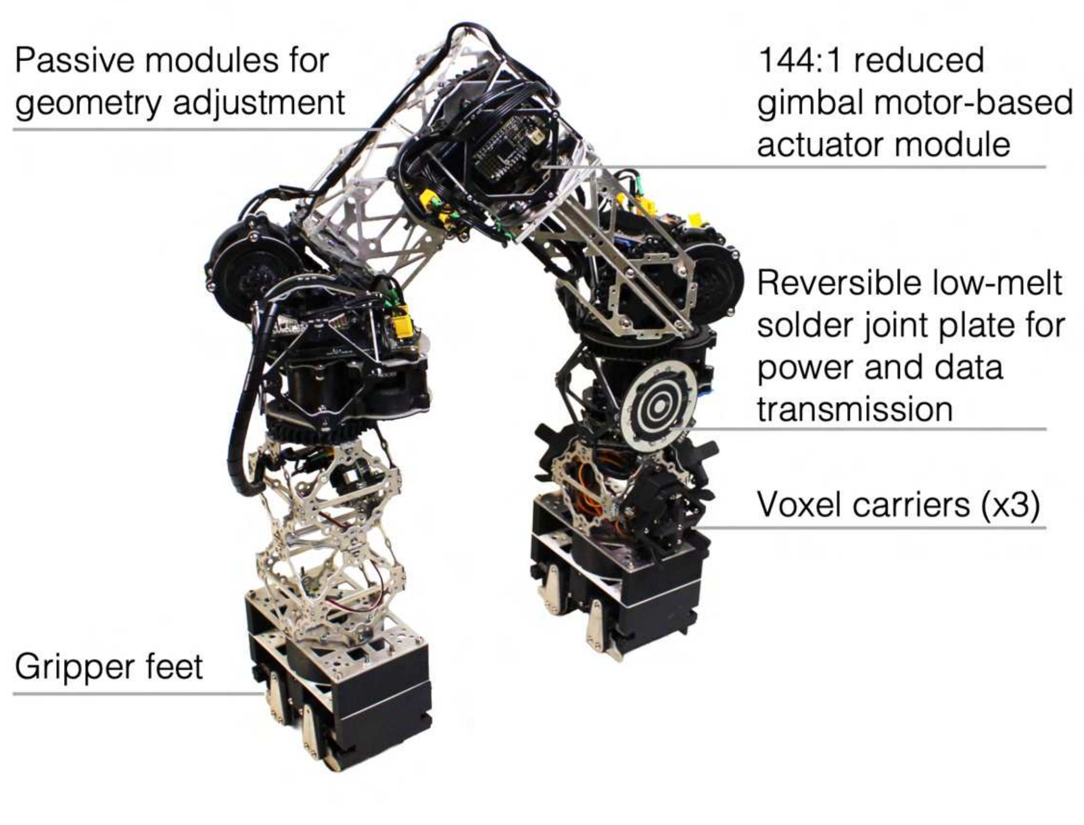

Figure 4-7: The Modular Inchworm Lattice Assembler robot (MILAbot) consists of four primary module types: actuated joints, passive spacers, gripper feet, and voxel carriers.

The MILAbot is made from four primary module types: actuated joints, passive spacers, gripper feet, and voxel carriers as showm in Fig.4-7. These modules are arranged in a configuration similar to a five degree of freedom robot arm, with grippers on both ends, and three voxel carriers on one side. Toward eventual robotic-self assembly, the modules are capped by incoming- and outgoing- PCBs on either side that use a heating circuit to reflow low-melt solder between modules, forming a simultaneous eletrical and mechanical connection, such as in [161], though this is not a focus of this work. The robot modules each have their own microcontroller, which are all networked over an I2C bus [4], with a WiFi-enabled microcontroller acting as the primary microcontroller on the bus. We typically run the robot at 14V, but the electronics are designed to handle up to 24V.

The actuator modules are built around a GM3506 gimbal motor, a small brushless DC motor. We use a two stage 3D printed reduction consisting of a 1:36 split ring planetary

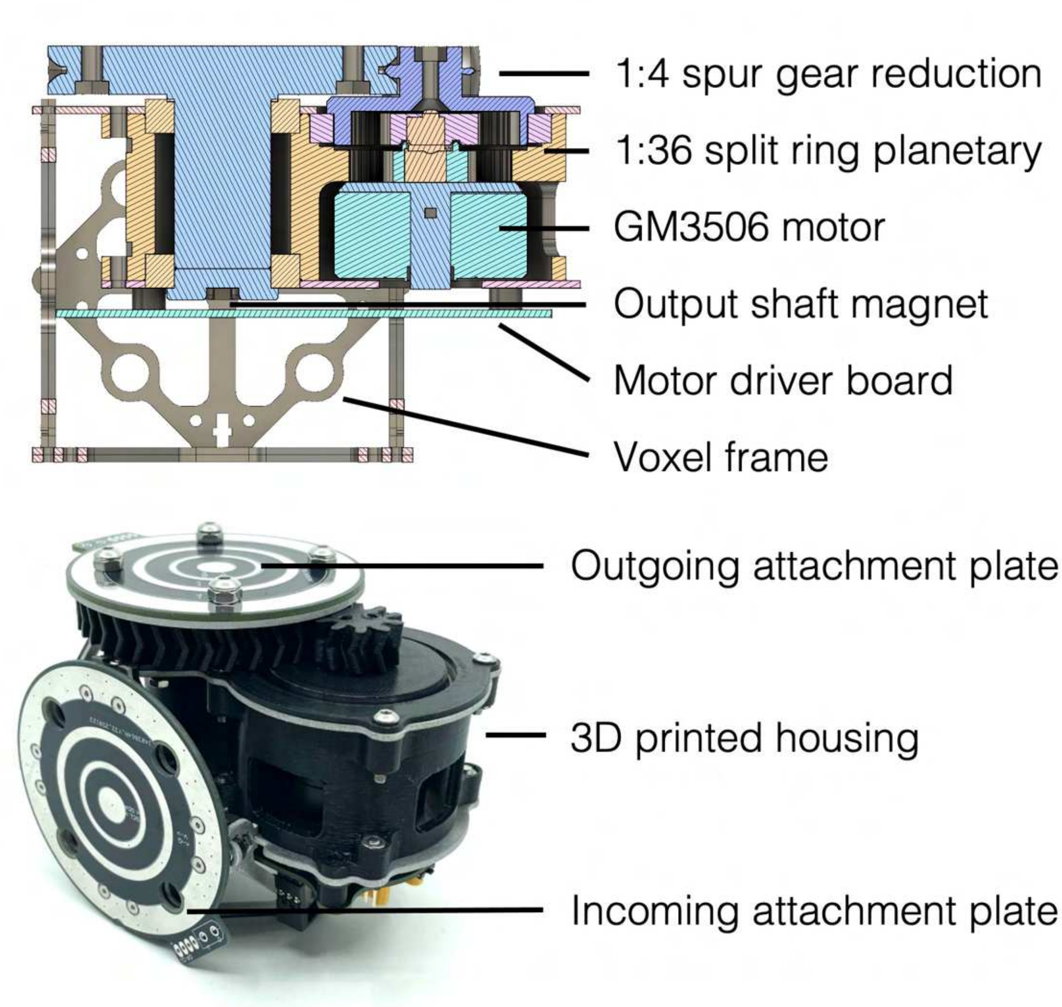

Figure 4-8: (Top) A cutaway view of the motor module design, which is based on a BLDC motor with two stages of reduction, feedback on the output shaft, and a voxel frame to enable module manipulation by another robot. (Bottom) A complete motor module, with incoming and outgoing PCB attachment plates.

followed by a 1:4 spur gear reduction, which has the added benefit of moving the output shaft off from the motor shaft, allowing us to more easily track the position of the output shaft. The total reduction is then 1:144. The module is fabricated from a mix of 3D printed polycarbonate, for its temperature resistance, and aluminum plates. The module controller board uses an Adafruit M4 Express (SAMD51) as the micronctroller, a DRV8316 for motor control, and two AS5407 magnetic encoders for motor and output shaft feedback. Motor control is done using SimpleFOC [160]. Each module weighs approximately 400g and can output approximately 10-15 Nm before failure.

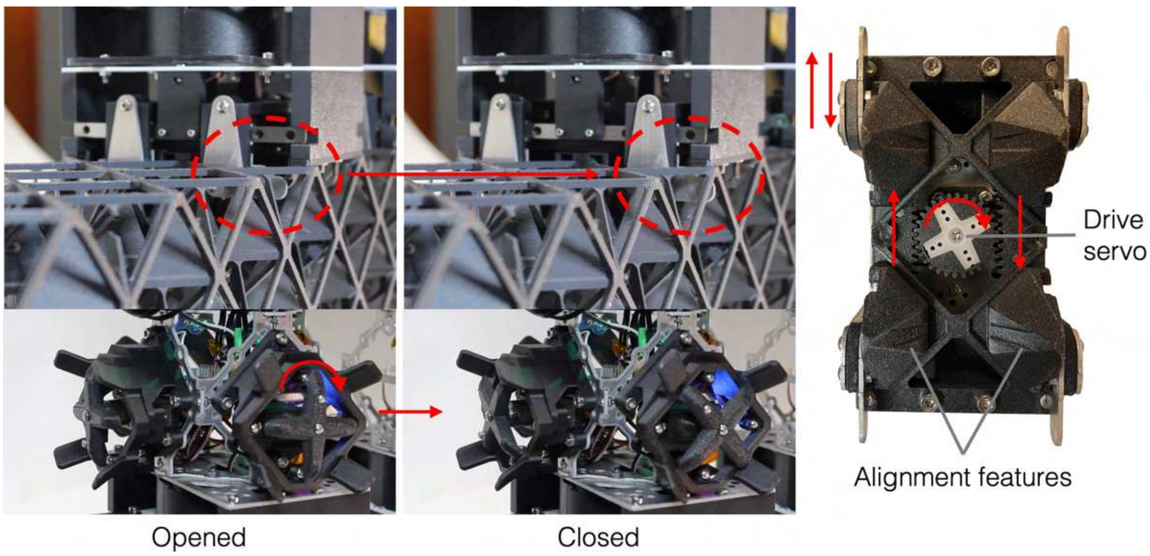

Figure 4-9: The operation of both gripper types in the MILAbot. (Top) The structure gripper, (bottom) the voxel payload gripper(s).

The MILAbot uses two voxel gripper types. The first type adheres the robot to the structure, and the second type carries a voxel payload. The operation of both types is shown in Fig.4-9. The gripper feet are designed to step in the middle of a 2x2x1 grid of voxels, which introduces some constraints on where the robot can locomote. Its relatively larger size affords the robot greater stability and helps to distribute the load of the robot's locomotion over a larger area on the voxels. Each gripper foot is actuated using a FeeTech FS117 hobby servo, which retracts and extends a set of aluminum gripping features underneath the upper corner nodes of some voxels.

Because the voxel carrier grippers do not need to support a significant load, they are designed to be smaller and lighter, using Miuzei MG90s micro servos to open and close a petal feature, modeled after the grippers used in [89] and [17]. To maximize the carrying capacity of the robot without significantly impacting its maneuverability, we use three per robot installed on one leg (though the robot could physically support installing an additional three voxel carriers on the other leg, this would cause significant collision issues for any non-flat structure).

Both grippers are designed with large alignment features. In the case of the gripper feet,

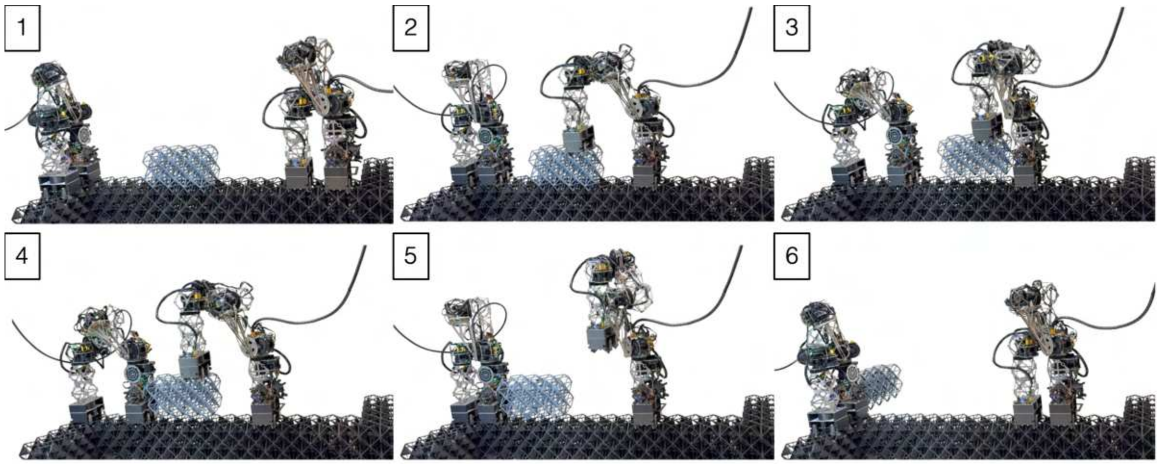

Figure 4-10: Picking up and passing a voxel block from one robot to another.

these match the alignment features of the voxel blocks themselves, while the voxel carriers use a smaller petal shape. The combination of these mechanical alignment features in the robot and in the underlying structure means that we are able to achieve relatively precise robot and voxel placement without high-performance robots (i.e. the robots have minimal feedback and a lot of compliance).

An example of this is shown in Fig.4-10, where two robots pass a voxel block to each other. The two robots approach the target voxel (gray), and one steps onto it and lifts to remove it (the voxel is not constrained with snaps in this case). The other robot then steps forward, re-indexing its position on the lattice. Because the voxel carriers are placed on the robot at a multiple of the lattice pitch, this places the second robot at a "known" location for the robot carrying the other voxel, which only needs to then move into that position, with the alignment petals proving fine alignment. The second robot then grips the voxel while the first lets go and retracts.

The robot relies on the ability of the voxels to correct large placement errors to install new blocks simply by dropping them into place and then stepping on them to engage the snap fits. This process is shown in Fig. 4-11, where a robot carrying two voxels steps into position, swings the back leg forward and rotates it to place the correct voxel in the appropriate pre-placement position, drops the voxel, returns the back leg backwards, steps forward once, and then stomps on the voxel to engage it, forming a sturdy connection.

4.7 Evaluation

In this section, we evaluate the performance of the presented system through a set of assembly demonstrations, the throughput and fidelity of the mechanical system, the mechanical efficiency of the lattice type, and general robotic performance metrics.

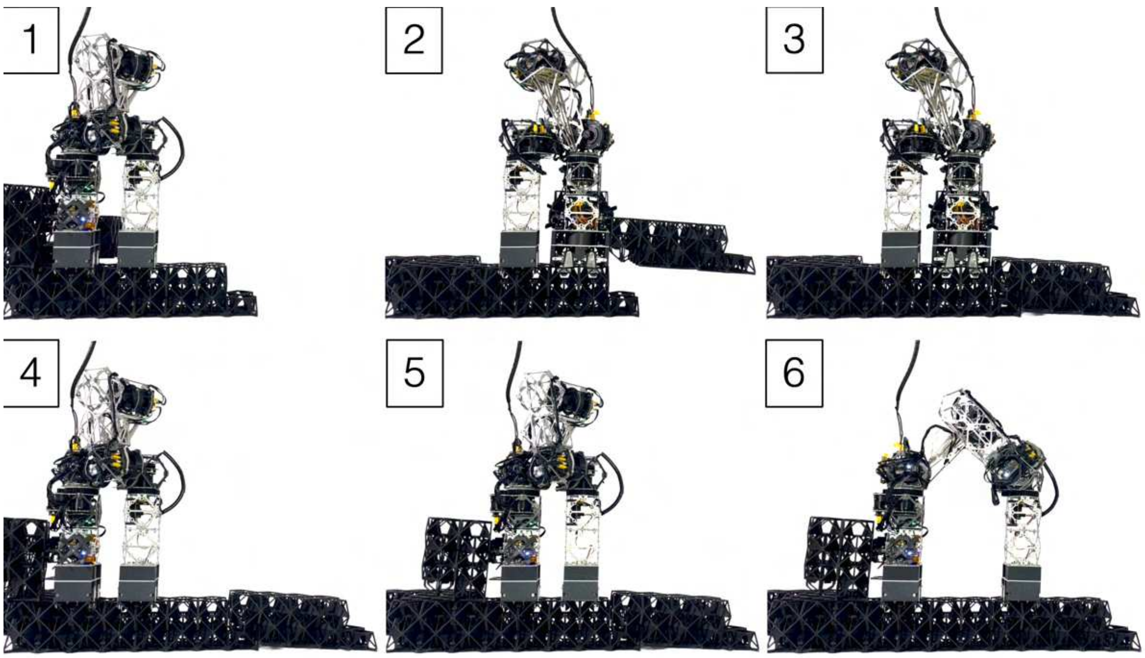

Figure 4-11: An example voxel installation sequence. 1) The robot is in pre-placement position. 2) The robot steps into the placement position and 3) drops the voxel roughly into place, before 4) retracting, and then 5) taking a step forward. 6) The robot stomps the new voxel to engage the snap fit connection.

4.7.1 Assembly Demonstrations

We use the complete workflow to demonstrate robotic assembly of basic structures using one and two robots.

Single Robot Assembly

First, we use a single robot to assemble a 4x4x4 voxel block. We voxelize the structure into four 4x2x2 blocks, the largest compounded size our system currently accommodates. This way, the robot only needs make four placements, in contrast to prior approaches to voxel assembly, which would have required 64 voxel placements for the equivalent structure. A condensed version of the assembly sequence is shown in Fig. 4-12. The robot carries two blocks at once, as the minimum amount of trips it can make for this structure is two (either two voxels twice, or three voxels and one voxel once). The first two voxels are placed oriented toward the camera plane, while the second two are placed parallel to it, resulting in a fully connected structure.

Next, each robot assembles its support staircase for the bench example, as previously shown in Fig.4-1 and Fig.4-4. Freeze frames from the build sequence are shown in Fig.4-13. The build process for the stair case is similar to that of the cube, but at the end, we demonstrate a small overhang with this structure that is able to support the weight of the assembler robot on it. Overhangs, or first layers, both require staggered voxel blocks to achieve. This can be done either by layer shifting a 2-voxel tall stack, or by attaching base-plate voxels at a stagger, as done here. The offset provides alignment features for the next block placement and braces the newly installed voxel against the existing structure, especially during the drop-to-place procedure.

Double Robot Assembly

Because each robot has its own voxel feed in our workflow, most multi-robot structure assembly is identical to single robot assembly, up until the point at which the structures meet. The joining of separate build fronts, or unlinked structures, is not something considered by any of the prior voxel assembly systems, but is likely a critical feature for scaling these systems up.

In this example, the two robots finish the assembly of the bench by filling in the base of the seat (see Fig.4-14 for reference). One robot drops a voxel block into the empty space, and then steps on it to align it and the structure. The second robot then installs its voxel into the remaining space. This is possible because the alignment features on the voxels are permissive of a fairly large amount of error, the error is larger than the voxels, and the structure is able to slightly slide and deform. We cannot always guarantee that this will be the case, so for joining build fronts, it is likely necessary to consider alternative voxel types, such as ones that are slightly undersized or compliant, to account for the potential

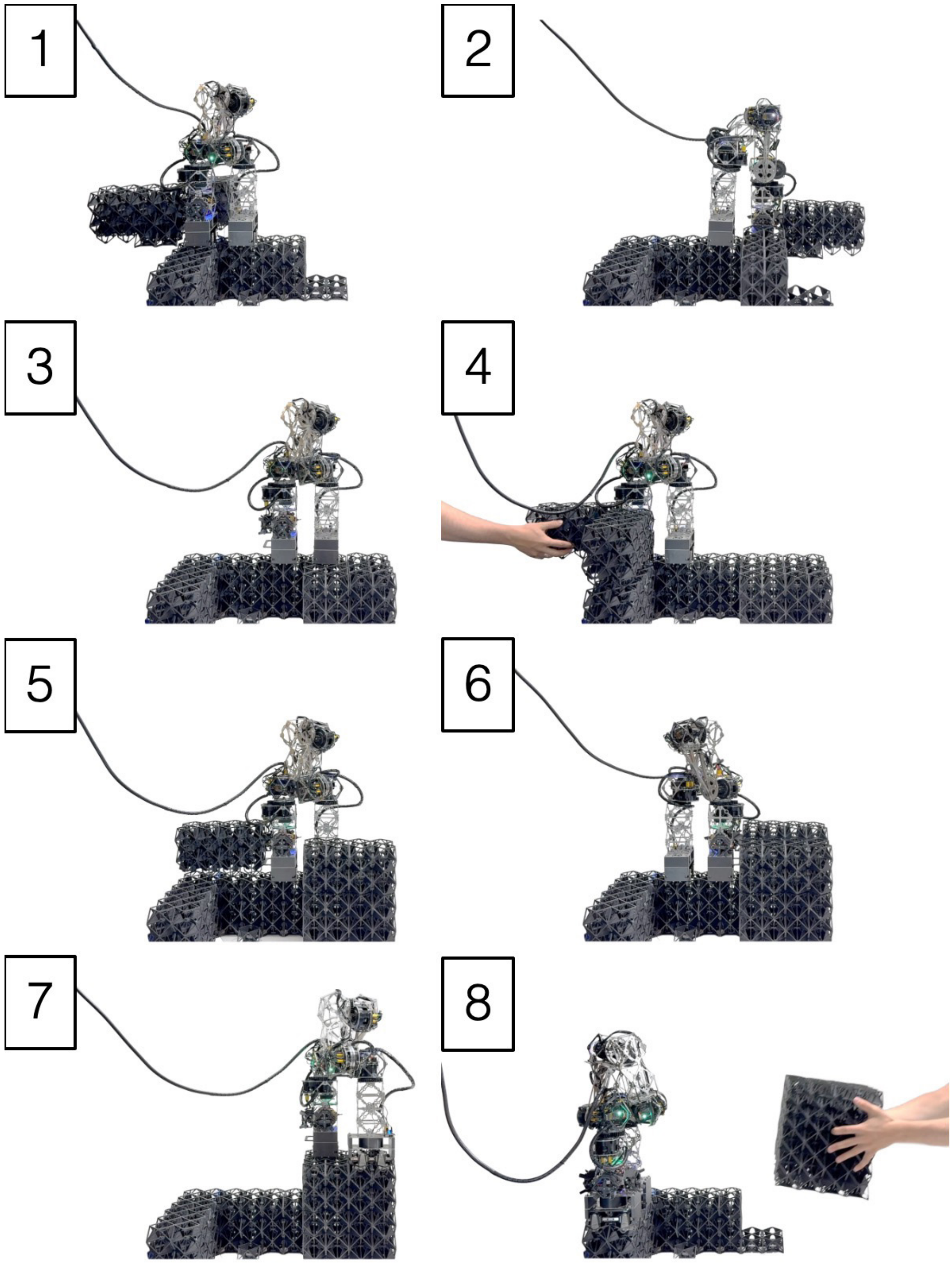

Figure 4-12: Assembly of a 4x4x4 voxel block using a set of four compounded blocks, carried two at a time.

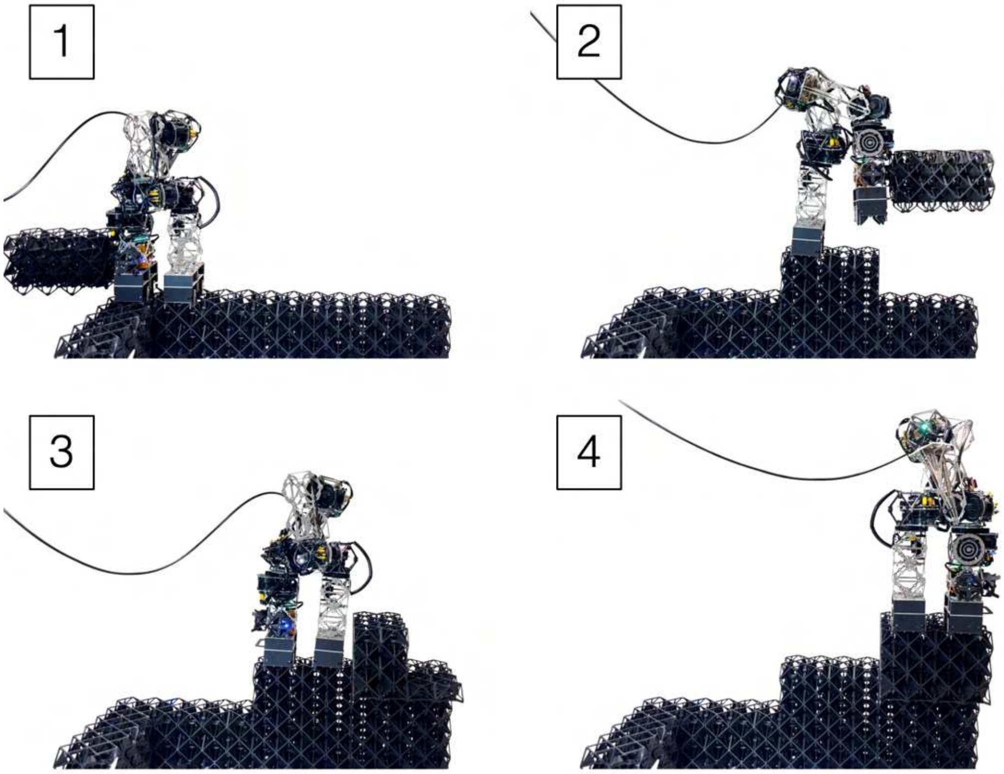

Figure 4-13: A single robot assembling a support staircase for the bench build.

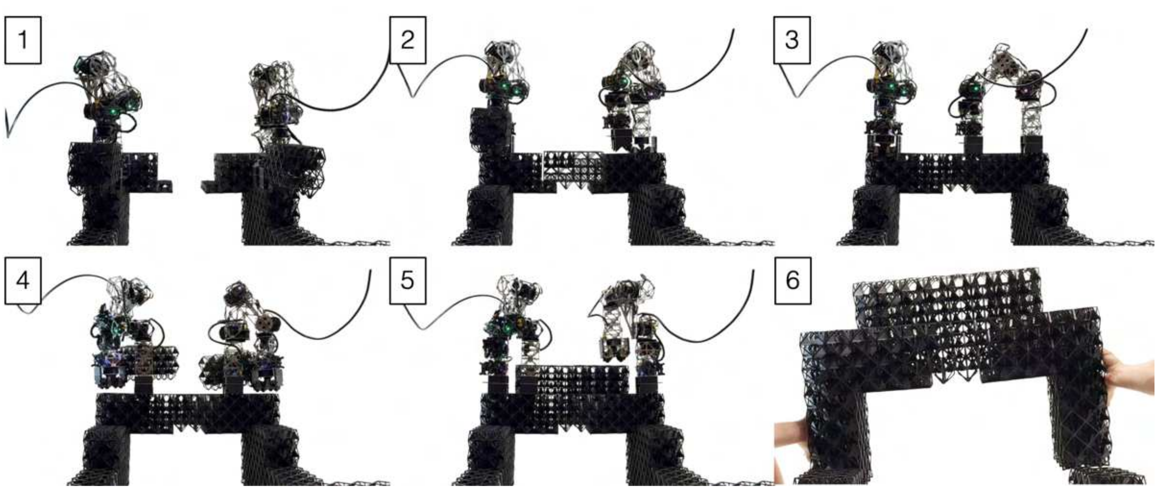

Figure 4-14: Two robots assembling a bench together.

misfit. However, even though this is not a reliable method for joining a voxel seam, this block placement is taken to be infeasible in the prior planning systems for voxel assembly [71], [161], [89].

After installing the base of the bench, the two robots install the back of the seat. These two voxels are necessarily installed simultaneously, as to avoid collisions in the placement. Though for a small structure with only two robots, keeping synchronicity (or restoring it if lost) is not difficult, as the system scales to larger robot counts, it may be necessary to consider additional methods for synchronizing the movements of all robots at critical junctures, without needing to run the entire system at the pace of the slowest robot. Or, the methodology for voxel placement could be revisited, to consider more sophisticated obstacle avoidance, depending on the build state of the structure.

Once completed, the bench is removed from the support staircase, where the path planning has left a seam. The completed bench is then directly usable, as the voxels have decent mechanical performance, which is further discussed in Section 6.3. We qualitatively evaluated the bench by having two people sit on it, as shown in Fig.4-15. The bench supports this weight, though the overall design is a bit small for comfort— luckily, the voxels support disassembly and reassembly, so we can change the design.

Figure 4-15: The assembled bench supports two people.

4.7.2 Throughput and Fidelity

As shown in Figure 4-16, we assess the performance of different voxelization strategies by analyzing the trade-off between the number of voxels that make up the structure and geometric precision across various mesh types. The geometric precision can be defined as:

$$precision = n_{voxel} * V_{voxel}/V_{mesh} \tag{4.2}$$

Where n_{voxel} is the number of unit voxel with V_{voxel} being the volume of one unit voxel and V_{mesh} the volume of the input mesh.

The left panel shows how larger voxel patterns drastically reduce the number of elements required but come with a loss in precision. The right panel highlights this trade-off by plotting precision as a function of voxel count. Patterns like 2x3x1 and 2x2x1 achieve a good balance, while the hierarchical combination offers a compelling middle ground with high precision and low voxel count.

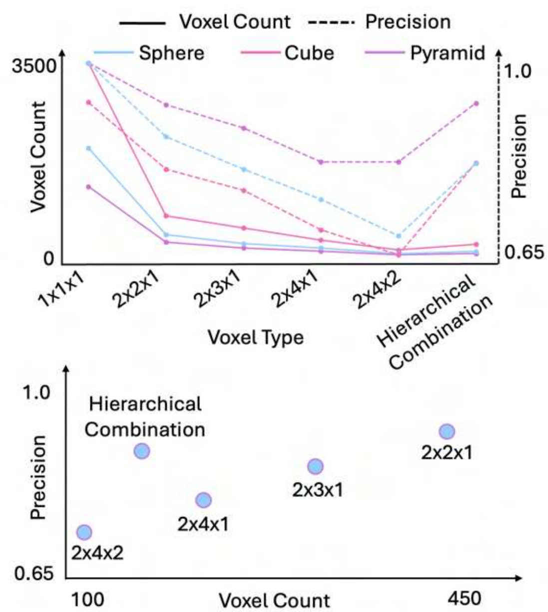

Figure 4-16: Evaluating voxel pattern efficiency— hierarchical combination outperforms any single voxel type by breaking the linear trade-off between voxel size and precision. (Top: Voxel count and precision decrease linearly with increasing voxel size when voxelizing meter-scale meshes. Bottom: Precision as a function of voxel count, highlighting the benefits of the hierarchical combination.)

Figure 4-17 highlights two key components to accelerate construction: increasing the number of robots and improving the voxel carrying strategy. On the top row, simulation snapshots show how multiple robots parallelize the construction of a 16x16x16 cube, reducing total build time. While the benefit is clear, the speed-up is not exactly proportional to the robot quantity due to each robot building its own support stairs independently, without sharing intermediate infrastructure.

The graph on the bottom left shows that construction time increases exponentially with structure size, but adding robots significantly flattens the curve. On the bottom right, we analyze the impact of voxel carrying capacity. Carrying multiple small voxels per trip already provides notable gains, but the greatest efficiency comes from transporting larger compound blocks (e.g., 4x2x2). This confirms that both scaling the number of robots and optimizing carried voxel configuration are critical for improving overall build throughput.

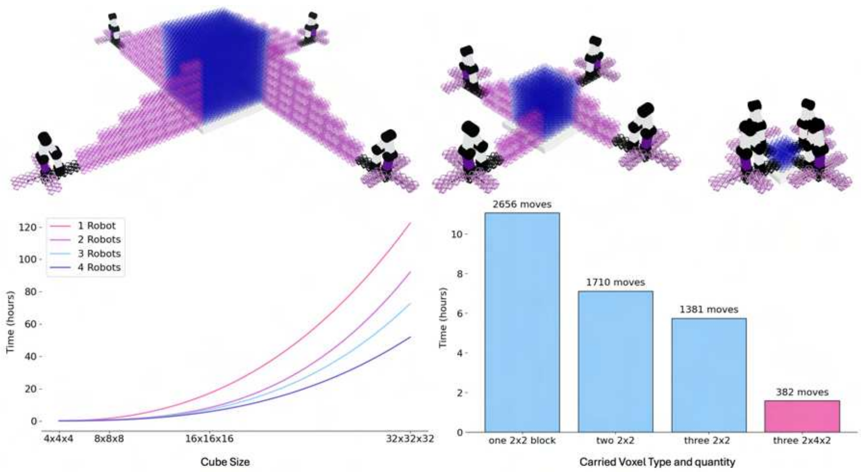

Figure 4-17: Impact of robot quantity and voxel carrying strategy on build time. (Top) Simulation snapshots of cubes of decreasing size (16x16x16, 8x8x8, and 4x4x4 voxels). (Bottom Left) Estimated build time as a function of cube size and number of robots. (Bottom Right) Time required to build the same structure depending on the number or size of voxels carried per trip. These graphs motivate the use of a larger robot fleet for assembling large structures, as they show a substantial decrease in assembly time. They also highlight the importance of maximizing robot payload, with each robot ideally carrying three 4×4×2 voxels rather than operating below full capacity.

4.7.3 Hardware Performance

Because other robotically assembled discrete lattices have been demonstrated, we can compare the performance of our system against those in literature. First, we consider the mechanical performance of the modified octet lattice we use. In evaluating this performance, we consider the version of the lattice presented here to be a proof-of-concept demonstration, and the comparisons below to indicate, generally if the design decisions we have made are in desirable directions. Because of the anisotropy in FFF printed parts, the robustness of our voxel system can be low, and we do not perform a full mechanical characterization of the voxels under tensile and bending loads. Further, the overall interplay between vertically interconnected blocks under various loads is also still unknown, and beyond the scope of the current work.

To determine the behavior of the lattice under compressive loads, we tested a single unit cell, a 2x2x2 lattice block, and a 3x3x3 lattice block on an Instron 4411 with a 5kN load cell for the single unit at the 2x2x2, and on an Instron 5985 with a 250kN load cell for the 3x3x3 at a compression rate of 10 mm/min. The results of this testing are summarized in Table 4.1. The lattices demonstrate good strength and stiffness at a reasonable weight (the average density of the lattice is 81.85 kg/m^3). Under compression, the lattices can support very high loads— the 2x2x2 block reached a maximum load of 3445N, which is about 2,220x its own weight, while the 3x3x3 block reached a maximum load of 8712N, which is about the load exerted by the weight of an average cow.

We can determine the mechanical efficiency of the lattice by looking at how its compressive modulus and density scale relative to the bulk material it is made from. The relative compressive modulus is given by E/E_s where E is the compressive modulus of the lattice and E_s is the compressive modulus of the bulk material. Similarly, relative density is given as ρ/ρ_s, where ρ is the density of the lattice and ρ_s is the density of the bulk material. Ideal stretch dominated behavior, that is, the most efficient scaling of modulus with density, is given by linear scaling E/ρ. However, this is generally considered inaccessible [156]. Instead, ideal bending dominated behavior is given by quadratic scaling $E^{1/2}/\rho$, while below $E^{1/3}/\rho$ is accessible via foams. So, the target for the structural efficiency of an architected lattice is to surpass the quadratic scaling for its bulk material.

We plot a comparison of the relative compressive modulus and relative density for this work, as well as the other discrete lattice assembly systems: [71], [161], and [89]. Both this work and [71] achieve stretch dominated behavior, indicating efficient and desirable material usage. Note that [89] does not provide mechanical testing data; this is instead estimated using an ideal beam simulation of the presented lattice using the reported parameters and materials. Additionally [89] uses a non-structural magnetic joint, but for this comparison we disregard that.

We can additionally compare the assembly throughput and system cost for these systems. We use the reported volumetric assembly throughput for each system, with the exception of [71], in which their reported throughput is for a hundred voxel structure, and so is an underestimate of the speed of the system as compared to the systems in [89] and [161] which only demonstrate assembly of tens of voxels. Instead, we draw an average from the installation time per voxel for the first 15 voxels in their structure, to better align with the data we have here, resulting in a much higher throughput estimate than what they report.

We find that our volumetric assembly throughput is 4,394,000 mm^3/min for a single robot carrying two blocks of 4x2x2 voxels, as compared to 2,700,000 mm^3/min for [71], 751,879 mm^3/min for [89], and 343,281 mm^3/min for [161]. Our higher volumetric throughput is owed to two main factors: we can carry multiple blocks of compounded voxels while moving at a comparable speed to [89], and unlike [71] the current implementation requires minimal installation/fastening time. This result indicates that we likely have budget to increase the time spent per voxel (such as by implementing a different connection system) while still staying competitive in volumetric throughput.

We additionally compare these results against the cost of each robotic system. For [161] and [89] these values are taken from their reported bills of materials, while for [71] we estimate the cost based on the components reported in [137], where the current figure is based on doubling the cost of the actuation system for one inchworm robot, to account for the need for both a carrying robot and an installing robot, with an additional \$300 added as a low-end estimate for the cost of the third robot, as well as all of non-actuation components in the inchworm robots. Interestingly, there is a relatively linear relationship between system cost and assembly throughput for the prior single-voxel based systems, which this work sidesteps via implementing hierarchical material handling in an otherwise very simple robot.

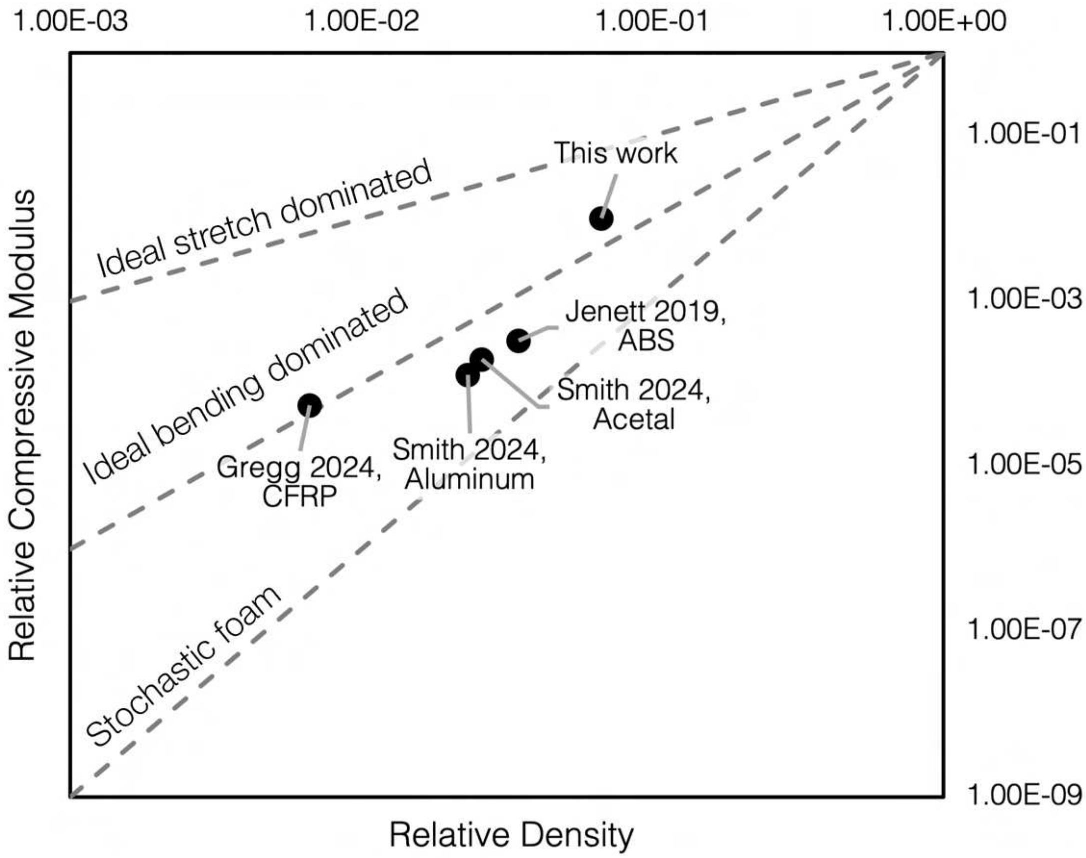

Figure 4-18: A comparison of the relative compressive modulus (voxel modulus divided by bulk material modulus) and the relative density (voxel density divided by bulk density) for robotically assembled discrete lattices. This work achieves stretch dominated behavior, demonstrating material efficiency relative prior robotically assembled voxel systems.

Table 4.1: Summary of mechanical properties for different voxel block sizes

	1x1x1	**2x2x2**	**3x3x3**
Stiffness [N/mm]	949	2278	4868
Maximum load [N]	602	3445	8712
Compressive Modulus [MPa]	14.6	17.5	24.9

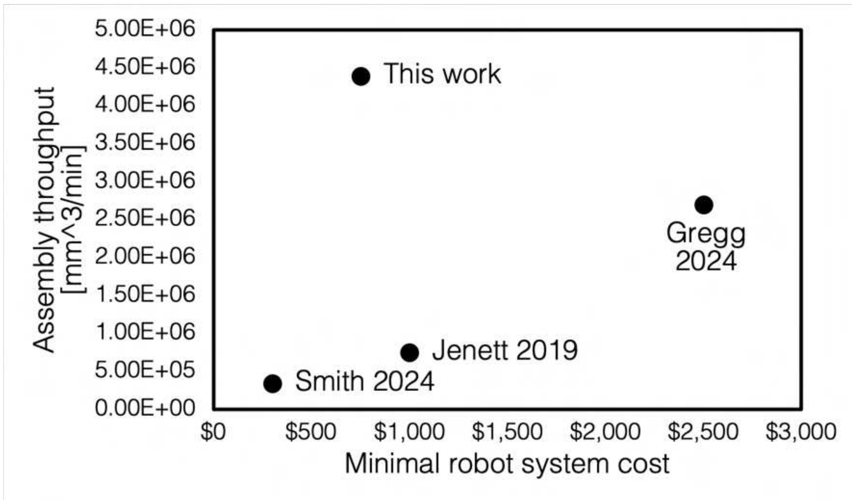

Figure 4-19: A comparison of the volumetric throughput for different voxel assembly systems for low voxel count structures, against the cost of the minimal robotic assembly unit. This work achieves the highest throughput at the second lowest cost.

4.8 Future Work and Limitations

4.8.1 Voxelization

While our voxelization algorithm proved sufficient to demonstrate the feasibility of a complete robotic assembly pipeline across a defined set of structures, it remains limited in its generalizability and is sensitive to variations in input geometry. In future work, we plan to explore more robust and generalizable approaches. This includes developing enhanced rule-based algorithms inspired by an octree subdivision approach as developed in [17], which remain fully discrete and deterministic, as well as investigating learning-based methods, such as those proposed in [148], that leverage the power of large language model to Generate physically stable and buildable LEGO® structures. Incorporating a structure-aware strategy that accounts for physical constraints and evolving geometry could significantly improve the reliability and feasibility of voxel placement over time. And would extend the buildable structure to the one presented in Figure 4-20. Future work will also explore generating input geometries based on desired performance, such by integrating the system in [107], or verifying structural performance, such as through built in FEA, as in [162].

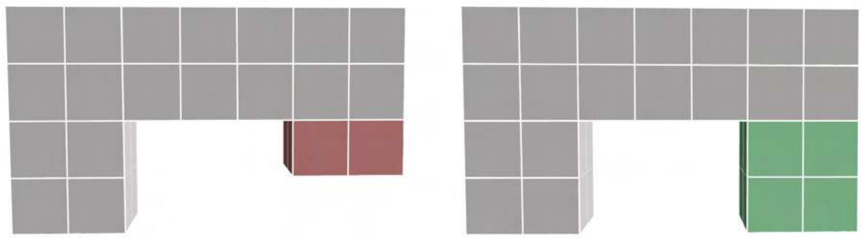

Figure 4-20: Comparison of infeasible (left) and feasible (right) structures due to local connectivity constraints.

4.8.2 Feeder Management

As discussed in Section 4.5.2, in the current setup, each robot is paired with a unique Feed location, allowing independent operation. This constraint simplifies coordination, but limits scalability and task flexibility. Future work will explore decoupling the number of robots from the number of sources, enabling more dynamic task allocation. Related work [17] has demonstrated that shared and flexible sourcing strategies can significantly improve resource utilization and system throughput.

Additionally, in the current system, we manually feed voxels to the robot. In a fully automated set up, this would be accomplished using e.g. conveyor belt to deliver voxels from their manufactured or stored location to the robots.

4.8.3 Robot Control

As the system scales to multiple robots, hardware variability is expected to introduce differences in kinematics and precision. To ensure consistent behavior across platforms, we plan to develop a more adaptable control system with integrated per-robot calibration. This includes implementing a tunable inverse kinematics (IK) model, allowing each robot to compensate for mechanical deviations and maintain precise control during assembly.

Additionally, to fully realize voxel reconfiguration, the robots will need to eventually be able to disassemble voxel structures, which will require both additions to the robotic hardware as well as the path-planning tool. Eventually, we hope to be able to fully support the arbitrary disassembly and reconfiguration of a voxel structure using this system.

4.8.4 Voxel Material

We use FFF printed voxel blocks in this work primarily because they are readily accessible and offer high performance along some loading directions. However, the anisotropies inherent in FFF plastic printing likely preclude its use for many practical applications, so future work will focus on developing alternate methods for pre-building the voxel blocks, such as through injection molded faces assembled together [91] or formed metallic facets [152]. Additionally, [162], [42], and [174] present electrically active voxel systems, which could be integrated with this system in the future. Additionally, the system is likely mechanically anisotropic, and in future work we plan to develop additional tools simulated structure behavior, as well as more fully characterize the material system under bending and tensile loads.

4.9 Conclusion

This work demonstrates a novel approach to scalable digital fabrication through the integration of architected lattice blocks, modular mobile robots, and a live digital twin system for coordinated assembly. By leveraging the strengths of small-scale digital fabrication to produce complex interlocking components, and combining them with simple, mobile robots capable of relative traversal and block placement, we enable the construction of meter-scale structures with reduced system complexity and cost. The hierarchical voxelization strategy, modular assembly robot design, and real-time simulation interface collectively contribute to a flexible and scalable assembly pipeline. Our validation of the system through the successful fabrication of meter-scale structures highlights its potential for broader application, such as for architecture, infrastructure, or in-space assembly and manufacturing. This work establishes groundwork for further exploring multi-robot collaboration in hardware, the interplay between structural robotic systems, and, eventually, hierarchical and recursive robotic assembly.

Chapter 5

Sustainability and Economics of Voxel Assembly for Construction

The work in this chapter appears in part in the publication "Comparative Evaluation of Robotically Assembled Discrete Lattice Systems for Sustainable Construction" and was co-written with my collaborators Alfonso Parra Rubio and Paul Richard. This was published in Automation in Construction in 2026.

5.1 Abstract

Additive manufacturing could improve the sustainability and efficiency of building construction, but current methods fall short. Continuous methods, such as 3-D printing, face challenges in error-correction, structural integrity, and accessible build volume, while assembly-based approaches typically require high mechanical precision, leading to costly and complex machines or off-site fabrication. This paper proposes robotically assembled discrete architected lattices, or voxels, as an in-situ construction method with low environmental impact and competitive speed and cost. Voxel-based structures can be scalably and incrementally assembled using distributed low-cost robots. The performance of multiple voxel types is compared against 3-D printed concrete, precast modular concrete, concrete masonry units, steel framing, and stick framing for a simplified 1-story building. The results indicate voxel-based systems can reduce embodied carbon by up to 76-82% relative to 3-D printed concrete, with competitive cost and construction timelines, demonstrating the potential of distributed building-block assembly as a sustainable route to automated in-situ construction.

5.2 Introduction

The construction industry drives 38% of global greenhouse gas emissions in energy [22], a significant problem, especially since construction demand is projected to continue to increase [54]. Digital fabrication has been proposed as a potential approach to mitigate the environmental impacts of construction by enabling more material-efficient designs with more diverse materials [66], while also improving the overall efficiency of construction, which is currently low [175] [37]. Although digital fabrication methods offer promise for improving the sustainability and efficiency of manufacturing in other fields, such as aerospace [99], they face additional challenges for building construction due to large structure sizes and variable building environments [187]. Managing the competing needs to balance precision and assembly throughput over large distances in relatively unstructured environments has led towards automation methods that are often large, complex, and expensive [122]. These systems rely on either very large machines, such as with concrete 3D printers [181], or repurposed industrial robot arms, such as in [172], to manage error accumulation, and handle heavy loads, but these fairly complex and inflexible automation systems introduce additional challenges that can inhibit broad adoption [122]. At the same time, while research in distributed robotic systems has demonstrated the ability to build large structures with relatively simple robots, many of these systems do not seriously consider how they might scale to practical real world building scenarios. In this work, we evaluate the feasibility of robotically assembled discrete lattices as an alternative approach to on-site building construction.

Our proposed approach is to use robotically assembled blocks of architected lattice material to form a material-efficient, robust, and automated building process for the structural elements of a building. Architected lattices are a method to achieve efficiently distributed mass in a given structure [156], similar to truss structures commonly used for the construction of bridges or space frames. To apply these systems beyond the micro- or macro-scale, a regular lattice can be decomposed into repeating unit cells, or voxels, which can then be assembled into much larger structures, maintaining the advantageous properties of the basic architected lattice geometry [46]. Recent work has demonstrated the robotic assembly of meter-scale load bearing voxel structures [71], while other recent work in [67] evaluated the use of voxels in a load-bearing wall versus other construction methods, demonstrating lower embodied emissions for the voxel approach. These results indicate potential promise toward using a voxel-based approach for building construction, where robotic voxel assembly could offer the following:

- Material efficiency and sustainability benefits due to voxel structural performance
- Reliable building tolerances through continuous error-correction during the incremental assembly process
- Efficient scalability through increasing robot quantities
- Low capital costs as the co-design of the robots and voxels enables a simpler robotic system

- A future route to fully reconfigurable structures

In this work, we project the performance of voxel-based building strategies against existing construction methods for a simplified one-story building. Based on the existing literature, we evaluated the mechanical performance and sustainability of eight voxel types, and we also designed and manufactured a new set of three voxel types compatible with robotic assembly. Using these, we evaluated the throughput and cost of a mobile robotic assembly system in hardware, to determine realistic time and cost estimates for voxel-based construction approaches. We compare these results against the performance of 3D concrete printing, precast modular concrete, concrete masonry units, cold-formed steel framing, and stick framing, across three main metrics: embodied carbon, material and fabrication cost, and building time. From this comparative study, we find that voxel approaches have the potential to reduce greenhouse gas emissions, cost, and assembly speed compared to existing construction approaches; however, the choice of lattice type, manufacturing process, and base material is critical to achieve any potential benefits. Our primary contribution is the evaluation of the efficiency of a robotically assembled voxel building system for construction in terms of sustainability, cost, and speed.

5.3 Background

In this section, we discuss current popular approaches to automating large-scale fabrication. We cover continuous extrusion-based processes, such as concrete 3D printing, and assembly-based processes using both static machines and distributed robotics.

5.3.1 3D Printing Systems

Of the continuous extrusion processes, 3D concrete printing, or 3DCP, currently commands the most research attention [115]. Multiple buildings have been demonstrated with this methodology, including multi-story buildings [56] and commercially available housing developments [60]. In a 3DCP system, a concrete mixture typically consisting of Portland cement and sand is extruded in layers, either through a gantry system or an industrial arm [187]. However, this approach faces many challenges. From a sustainability perspective, material savings promised by the freedom of additive manufacturing are often not realized or are minimal [16], and the composition of the concrete used is typically more carbon-intensive than traditional concrete [30]. Although some studies have shown that 3DCP buildings have lower embodied carbon values compared to standard reinforced concrete buildings [16], [30], this is derived in large part from the lack of steel reinforcement, negatively affecting structural performance. Besides, this contrasts with the predominant commercial use of this technology, which uses the 3D printed structure primarily as a stay-in-place formwork for traditional reinforced concrete [97], [34], which results in worse sustainability outcomes because of the inefficient use of material. Additionally, many concrete printing blends have higher embodied greenhouse gas emissions because of the higher cement and additive concrete blends needed for printability [67].

Beyond these issues, concrete structures cannot be non-destructively changed, such as running new utilities through a building or changing interior wall layouts, and the material cannot be meaningfully recycled [52]. Although concrete structures in general are often cited for their durability [166], due to challenges in material formulation and inter-layer bonding, these benefits may not fully extend to current 3DCP approaches [128]. And, though there is currently research on methods for reinforcing extruded concrete, such as [118], these still face significant challenges before deployment, which limits the application of 3-D printing primarily to compression-only components. Even then, it is common for 3DCP to simply be used as formwork for standard reinforced concrete [34], further lowering the value proposition of the system.

Due to these challenges, there is active work on printing with more sustainable materials, such as with earthen mixtures [52]. WASP, for example, has demonstrated multiple buildings printed using a locally produced earth mixture [12]. This work is still limited by the lack of reinforcement, resulting in materials-intensive buildings, which can negate some sustainability benefits [52]. Unlike standard concrete, these structures offer the potential for recycling the base building material, leading to a more circular lifespan [52].

Various other projects have explored printing beyond cementitious materials, such as plastics and metals. [5] has demonstrated a bio-based polymer single-story house for research purposes, while [145] has demonstrated multiple spanning pedestrian bridges made of thermoplastic. [8] has demonstrated off-site printed truss lattices for multiple architecture-scale pavilions, sculptures, and cladding. Additionally, [10] demonstrated a 3-D printed welded steel pedestrian bridge, and on a similar scale, 3-D printed metal rocket parts have been demonstrated [6]. In particular, in these projects, the fabrication of the structure is done off-site and later delivered to the site. This is done as the complete structure, in the case of [10], or as large modules, as in [145], which are then installed in situ. This represents a similar workflow to prefabricated modular building units, but without the benefits of mass-production [114].

5.3.2 Assembly-Based Approaches

Other large-scale construction platforms have explored assembly-based approaches. Bricklaying has been automated by multiple groups, such as [11], which uses a robot arm on a mobile base to place clay bricks, or [13], which uses a telescopic 32-meter robotic arm to place mortar-less concrete masonry units (CMUs). Timber frame assembly has been explored using mobile robots as well as fixed gantry style assembly, such as in [59], [124], and has been commercially deployed by [9] for load-bearing framing. More uncommon material systems have also been demonstrated, such as filament winding in [38]. Although these projects avoid the challenges of concrete deposition, they are still limited by either the need for high human intervention, an extremely well-defined build area, or highly expensive machines. These limitations stem from the challenges of maintaining accuracy over building-scale sizes— that is, error in the placement of a single element, such as a brick, impacts future placements and compounds on any prior errors.

Across these approaches, there is additionally a clear challenge in scaling beyond one- to two-story buildings, in that the operating envelope of the building system limits the ultimate size of the assembled structure. Beyond simply building vertically, these approaches would also face challenges in any project with an overall large footprint, especially for buildings or structures in difficult to access areas that cannot support large machines, such as for infrastructure development [123]. This is especially true for gantry girder systems, which are typical of current commercial demonstrations of 3DCP. Though this issue can be partially mitigated by using an industrial arm on a mobile base, or by moving the anchoring of a gantry printer, this presents multiple challenges of path planning and re-indexing position [182], which has so far limited the performance of this type of system.

Prefabrication of larger modules also presents an opportunity to leverage digital fabrication or manufacturing processes on the construction site. Fully prefabricated housing modules, such as [47], have existed on the market for some time, but are limited in their ability to be customized and easily transported. Panel-based approaches alleviate many of these challenges, and are available in a range of material types and applications. Pre-cast modular concrete systems are relatively common for large commercial and industrial applications and in smaller projects that do not require a high degree of customization, such as for basement foundations [83]. Other methods, such as tilt-up slabs [170] or stay-in-place insulated concrete formwork [85] also enable a form of overall increased automation. The size and weight of these concrete systems can make installation difficult, and can make attempts at automating these processes more difficult.

Wood-based modular systems are also common and avert some of the challenges of the modular concrete systems. These can span a wide range of finishes, complexities, and applications. At the simple end, structural insulated panels are common in residential construction [127], while many companies provide more finished panel systems as well. Some of these more directly leverage robotic control or digital fabrication in their production processes to enable easier customization [109]. Using digital fabrication processes during the production of these larger modular panels can additionally help enable future automation of their assembly. [135] for example, explores using material-integrated assembly features to enable automatic assembly via a heterogeneous robotic system toward high performance timber buildings. By moving the production of these systems on site in structured environments, these systems can further avoid challenges associated with transporting large structural elements (and designing them to survive transport). This can be achieved through the development of systems akin to microfactory modules [172].

Other approaches to automation on the construction site involve the direct automation of existing equipment or processes for robotic operation. These approaches often address specific subtasks of the construction project which potentially limits their ability to achieve full autonomous function [122]. Commercial examples of these types of systems include [40] for bulldozing, or robots for drywall hanging [43].

5.3.3 Distributed Robotics

Distributed robotic construction systems have been demonstrated to build structures much larger than the individual assembler itself [144]. In a distributed robotic construction system, many mobile robots individually pick and place new material to build a structure, and thus avoid constraints on the building envelope. Although various aerial drone systems have been demonstrated, such as in [28] with foam bricks, [125] with cables, or [69] with specially-designed concrete masonry, these face additional challenges in path planning, coordination, and payload weight limits for flying systems. Alternatively, robots that locomote over or through the structure are also common, such as [142] , where robots stack magnetic bricks, [78] where micro-robots assemble carbon fiber trusses, or [112], where robots stack wood framing. The performance of some of these systems may be improved by considering how they can operate in conjunction with larger robotic systems, as more multi-scale and heterogeneous robotic systems [135]. Examples of this include [159], which explores using a relative robot to lay continuous filament into larger wood panels toward the architectural scale, and [113] which explores using a robotic clamping system in conjunction with industrial robot arms for architecture scale assembly demos in a lab environment.

Though many distributed robotic construction systems have been demonstrated, most of these demonstrations are on a relatively small scale and have very limited mechanical performance. However, some systems have demonstrated the ability to support spanning structures. Specifically, [161] and [71] demonstrate robotically assembled voxel systems that are load-bearing. While [161] only demonstrates this for centimeter-scale objects, [71] demonstrates a meter-scale structure assembled by a 3-robot team over several days.

5.3.4 Architected Lattices

Distributed robotic assembly of discrete elements faces an inherent challenge in terms of balancing unit count, resolution, and structural integrity against each other. While using smaller assembly units leads to better geometric resolution and lowered demands on the robotic assembly system, this leads to a larger amount of units needed for a given structure, and with that, a higher number of joints between units, which introduce further opportunities for assembly error and can reduce the overall strength and stiffness of the final structure. For this work, we explore using discretely assembled architected lattices because these systems are specifically designed around minimizing the impact of the joint systems on the overall structure's behavior.

Prior literature on architected lattices has established that these structures can achieve better stiffness and strength scaling at light weights [46], though much of this prior work has utilized micrometer to millimeter scale fabrication processes, such as two photon lithography [184], which limits the application of these types of structures. However, the behavior of these lattices is largely governed by their geometries, and so is relatively scale-agnostic. This means that physically larger versions of these lattices assembled from discrete components can achieve equivalent mechanical properties using scaled-up versions of the same geometries. This is demonstrated in [46], which used interlocking faces made from oriented carbon fiber to form a node-connected octet lattice with lattice pitch on the order of tens of millimeters. [46] demonstrated superior stiffness results in the ultra-lightweight regime compared to continuously manufactured microlattices. Similarly, [72] demonstrates comparable performance to hollow-tube metallic microlattices using injection molded fiber reinforced plastic voxels with a 75 millimeter lattice pitch. Prior work has demonstrated comparable or superior performance to continuously produced microlattices, such as the metallic and polymer microlattices described in [157] or [184], while spanning a range of unit cell sizes from 15 mm [95] to 300 mm [71]. These systems have been used to assemble structures on the single meter scale, such as bridges [92], furniture [162], aerostructures [51], or vehicles [91].

These macro-scale assembly-based architected lattices achieve equivalent behavior to their continuously manufacturing micro-scale counterparts because in both cases the overall behavior is dominated by the lattice geometry, and not by the connectors between the discrete components. This is achieved by carefully designing the connections such that they are very stiff relative the beam elements, while adding minimal weight, which can be achieved through a variety of connection types and materials through considering the overall load path. [46] for example, interleaves the structural face components to constrain all but one degree of freedom, with a secondarily installed clip acting as a shear constraint. [72] directly transmits loads through the connectors, but uses steel screws and nuts, which are substantially stiffer than the FRP beams, such that the overall behavior is dominated by the beam geometry. These joints are also typically designed with mechanical alignment features that enable a higher degree of alignment between components, such as the quasi-kinematic surface features from [72], [92], or [71]. Larger assemblies additionally benefit from an elastic averaging effect that lowers the overall error of the structure as well [94].

Most of the prior discrete architected lattice systems have been manually assembled, limiting their overall applicability. However, the robotic systems in [161] and [71] are designed to assemble load bearing architected lattices out of repeating unit cells, or voxels. The goal of this is a building system where the assembler size is decoupled from the ultimate structure size. Robots and voxels are designed to mechanically error-correct, so relatively simple robots may be used to achieve precise assembly [89]. In this way, many of the challenges associated with building in free space are circumvented, as the robot operates only within the well-defined space of the voxels and relies significantly on mechanical alignment features to navigate and locomote. The system presented in [71] , for example, manipulates blocks at a similar size to the bricks in the commercial HadrianX [13] bricklayer system, but can navigate and build without a real feedback system, while the HadrianX system is built up around a proprietary large-scale multi-DoF stabilizer, similar to the robotic arm on a crane system in [101]. Although these prior voxel assembly systems were primarily developed for aerospace applications, they also have potential for building construction. [67] adapted the voxel system from [91] for use with a partially bio-based resin reinforced with recycled carbon fiber, and compared the resulting voxel's performance against some existing construction approaches, demonstrating a lower embodied carbon value. In this study, they only consider a load-bearing wall and use a manually assembled voxel feedstock. However, one of the main appeals of the voxel system is the potential ability to construct overhanging elements, such as roofs, using on-site using robots. So, there is still substantial space to more thoroughly evaluate the performance of voxels for a construction application. These prior results suggest that the assembly of voxels using relatively simple and cheap robots may represent a mass-efficient (and therefore more sustainable) approach to construction without requiring large machine or robotic systems.

5.4 Methods

We proceed to compare 3-D printed concrete, precast modular reinforced concrete, concrete masonry units, cold-formed steel framing, and stick framing against multiple voxel-based approaches for a simplified single-story structure. The construction consists of four walls and a roof with length 10 m, width 5 m, and height 3 m, as shown in Figure 5-7. To achieve a similar level of finish in each building approach, we add external and internal sheathing to the framing and voxel based approaches. For each approach, we also add insulation for equivalent thermal performance.

Based on these parameters and finishes, we generate estimates for material consumption, fabrication time, overall cost, and embodied carbon for each of the building approaches. We evaluate the embodied carbon using the kilograms of CO_2 equivalents (CO_2e) for each approach, considering the production of the raw material, the processing of the building material, and the machine use in the construction process. The following subsections provide a summary of each building approach as well as a description of the hardware developed for this work. The embodied carbon and cost figures are provided in Table 5.3 in the Appendix.

5.4.1 Voxel Approaches

In this subsection, we assess the feasibility of using voxels to construct a building. We begin by comparing the mechanical performance and sustainability of voxel designs reported in the literature. We take results from this evaluation to inform the design of a new set of voxels and a robotic assembly system. Finally, we describe the key parameters of the evaluated voxels, the details of the newly developed designs, the robotic system, and the methods used to derive the CO_2e emissions, cost, and assembly times for each approach.

Voxel Types

We evaluated eleven types of voxels for their suitability for use in building construction; eight of these are from the literature, and three are developed for this work. With the exception of [67], much of the prior voxel research has focused on applications outside of construction, so the systems are not necessarily optimized for the requirements of building construction. Additionally, robotic assembly has not been demonstrated for the majority

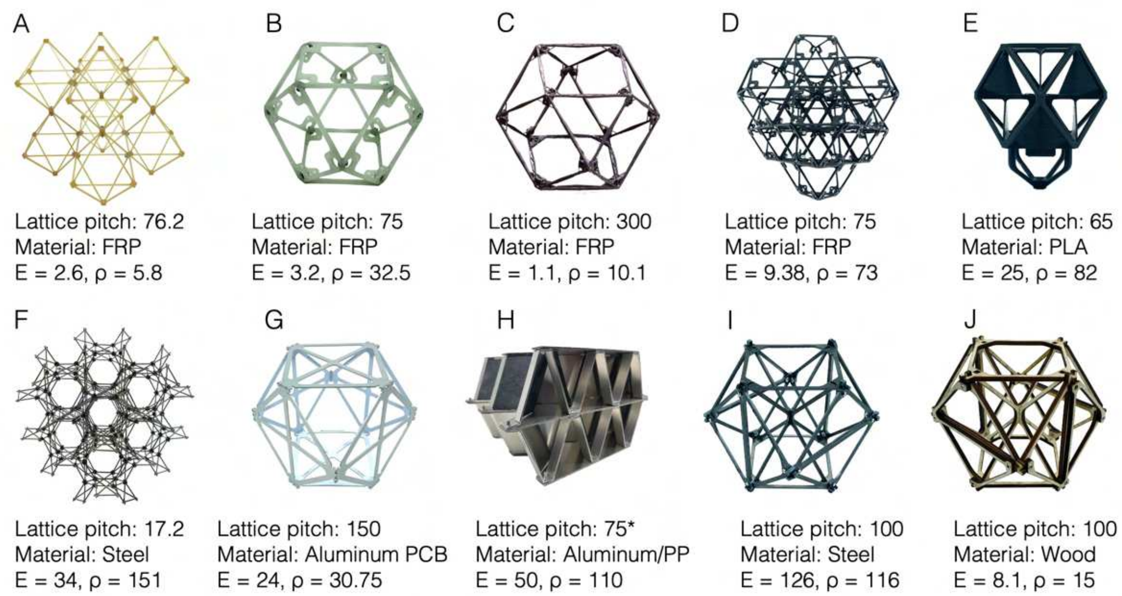

Figure 5-1: Summary of the considered voxel types, indicating lattice pitch in mm, base material family, compressive modulus E in MPa, and density ρ in kg/m^3.

of the evaluated voxel types, except the voxels developed for this work and the voxels from [71].

The evaluated voxels span several lattice types and multiple material and manufacturing families. These are summarized in Figure 5-1, which additionally notes the lattice pitch in millimeters of each voxel (that is, the side length of the bounding box around a single unit cell of the lattice), its general material family, and its compressive modulus and density. We provide a summary of the lattice geometries, materials, and manufacturing methods in this section, while a detailed description of how the material costs and CO_2e emissions were found is available in 5.8 and summarized in Table 5.1.

Figure 5-1A shows a 2x2x2 block of node-connected octets from [72] made from polyetherimide (PEI, brand name Ultem 2200) plastic with 20% glass fiber reinforcement. [72] additionally developed a 50% carbon fiber version (RTP based) in the same geometry, which is not pictured, but is considered in this work. Both are made via injection molding of the entire octet cell. We refer to these as the "GFRP node connected octet" and "CFRP node connected octet" hereafter.

Figure 5-1 B, C, and D are all face-connected cuboctahedra ("cuboct"). Figure 5-1B is made of 30% glass reinforced nylon 66 (Zytel 70G33L). We refer to this voxel as the GFRP cuboct. 5-1C is made from 50% carbon fiber reinforced nylon (Stat-tech NN), which we refer to as the CFRP cuboct. Figure 5-1D ("bio-cuboct") was developed in [67] as a more sustainable version of the GFRP cuboct from [91], and is made from a partially bio-based PTT with recycled carbon fiber reinforcement.

Table 5.1: Summary of carbon intensity, specific modulus, unit cost, and cost relative to compressive modulus of all voxel types.

Voxel Type	**Carbon intensity** $kgCO_2e/kg$	**Specific modulus** MPa/$\frac{kg}{m^3}$	**Unit cost** USD/kg	**Unit cost per modulus** USD/kg/MPa
Aluminum Cuboct (G)	30	0.78	14	0.58
GFRP Node-octet (A)	12	0.45	40	15.38
CFRP Node-octet (A)	19	1.37	172	21.58
GFRP Cuboct (B)	5	0.10	385	120.31
CFRP Cuboct (C)	13	0.11	79	71.82
Steel Kelvin (F)	4	0.23	13	0.38
Bio-Cuboct (D)	5	0.13	210	22.39
Expanded Miura (H)	9	0.45	28	0.56
PLA Octet (E)	3	0.30	22	0.88
Steel Octet (I)	2	1.09	7	0.06
Plywood Octet (J)	1	0.54	8	0.99

We evaluate three metals-based approaches from the literature. Figure 5-1F depicts a reinforced Kelvin lattice made from laser-cut mild steel from [95]. We include this lattice as an example lattice made from purely 2D components which does not consider efficient material usage in its production. Figure 5-1G is of a reinforced aluminum cuboct lattice with integrated electronics from [162]. We include this as an example of a structural voxel system with integrated electronics, to help evaluate the feasibility of this approach for applications that may benefit from electronics functionality. Figure 5-1H shows a discretely assembled plate lattice from [139]. Unlike all of the other evaluated voxel approaches, this approach is based on plate lattices, as opposed to truss lattices. The lattice is formed using an expanded Miura pattern and made from folded aluminum-polypropylene sandwich material (Hylite, 0.2 mm thick aluminum on either side of a 0.8 mm thick PP core) cut on a large-format CNC cutter.

For this work, we developed a new voxel geometry aimed towards easier robotic fabrication that we produced using FFF-printed PLA, laser-cut mild steel, and laser-cut plywood. The PLA version aims to facilitate prototyping, while the steel and plywood versions are aimed at providing suitable mechanical performance using more traditional construction materials. The PLA version is shown in Figure 5-1E, while the steel and wood versions are in 5-1I and 5-1J, respectively. These form an edge-connected octet lattice, and are discussed in more depth in the following section.

Voxel Designs for Robotic Assembly

We developed the three voxels shown in Figure 5-1 E, I, and J for compatibility with robotic assembly. The PLA octet is largely intended as a prototyping tool for convenient use to test

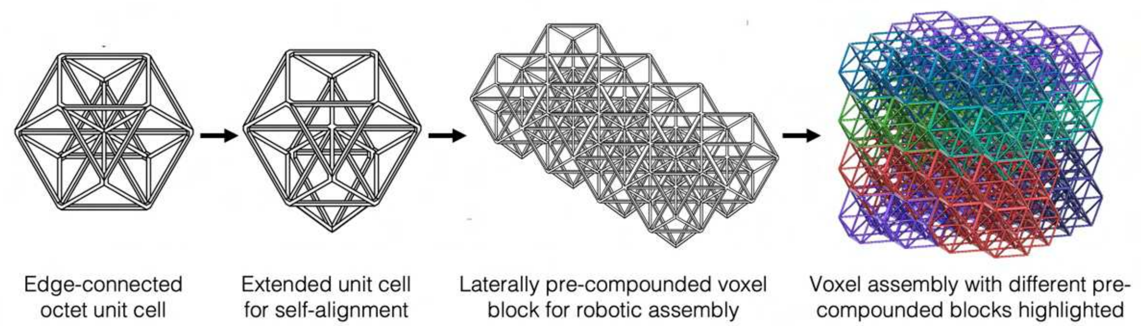

Figure 5-2: Modified edge-connected octet lattice for voxel design.

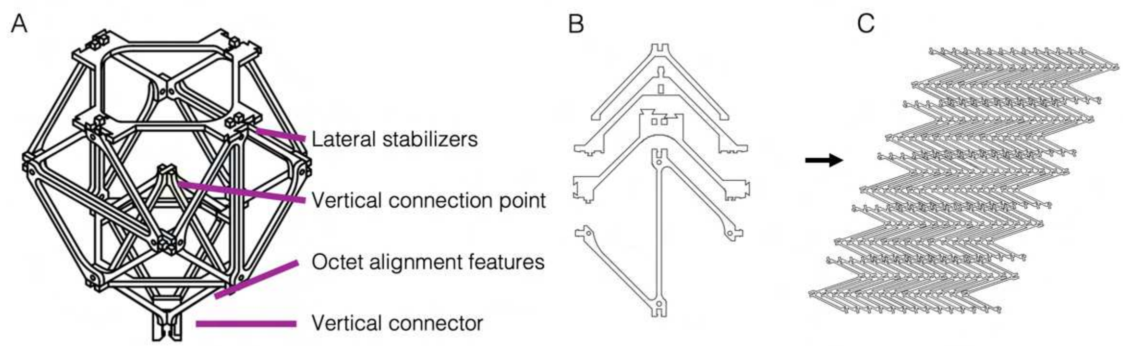

Figure 5-3: Efficiently packable 2D decomposition of the extended edge octet unit cell.

the robots with, while the steel and plywood versions are intended as efficiently packed 2D decompositions of the lattice more appropriate for potential future use in large structures.

This new design, intended for robotic assembly, exploits the geometry of semi-regular spatial tilings to form structurally-active self-alignment features. Each face-connected cuboctahedral cell can also be interpreted as face-centered cubic (FCC) octahedral voids. Figure 5-2 shows our approach using this property as a self-alignment mechanism: by shifting the FCC octahedral void downward by half its height, we generate self-aligning legs for the building blocks. From there, the unit cell is pre-compounded into larger blocks (such as the 2x4x1 block shown, or staggered block elements) that can then assemble into larger structures, such as the 4x4x4 cube shown, with different pre-compounded elements highlighted in different colors. To facilitate faster robotic assembly, these extended fabrication unit cells are pre-compounded into larger blocks, such as the 2x4x2 block shown in Figure 5-2, or into staggered blocks, so that they can tile into arbitrary 3D shapes, forming a fully connected structure.

The interlocking voxel geometry constrains every degree of freedom except for the installation direction. There are many possible options for constraining the final degree of freedom, in this case the vertical connections, but for this design we use a snap fit connection at the base of the extended octet, as shown in Figure 5-3. Lateral stability is provided by a combination of the nested octet as well as extended dove-tail features along the upper plane of the voxel. This joint system is designed to enable a load-bearing connection without needing the robot to access the connection plane or location, which improves assembly speed. However, it is possible to use a variety of other options. For example, [71] uses a custom symmetric rotational connector that is actuated by a robot internal to the lattice, while [72] uses standard screws, [91] uses rivets, and [95] uses a shear clip. These joints are designed such that their overall impact on the behavior of a lattice assembly is minimized, and this nesting octet lattice decomposition enables loads to be effectively transmitted through the structure despite partial connections between lattice blocks, which can seen in the bending behavior of some of these assemblies, as shown in the Appendix 5.9.

The nesting octet-cuboct structure creates cascading envelopes of mechanical error correction that force newly placed voxel blocks into the correct position, with the overall error governed by the manufacturing tolerances of the original lattice production method. In the horizontal plane, the design permits errors up to $\frac{p}{2} - 2 * t$, where p is the lattice pitch, and t is the beam thickness, assuming constant thickness beams throughout the voxel. For the PLA voxel, this value is ± 25.5 mm, for the steel voxel it is ± 44 mm, and for the wood voxel it is ± 42 mm. Because the voxels can be compounded into blocks of different shapes, the exact amount of permissible horizontal-plane angular error varies, but for these three designs, errors within ± 9 degrees from the longitudinal pivot of the endpoint of this line are allowed. These provide large boundaries within which voxel blocks can be successfully and repeatably attached. Additionally, the simultaneous placement of multiple unit cells at once, and as the structure gets larger, these voxel systems will benefit from an elastic averaging effect that can overall reduce the error associated with a given build [92].

To facilitate easier fabrication, the designs of the plastic, steel, and plywood octets diverge from the basic octet geometry. In the case of the PLA voxel, because it is FFF 3D printed, flanges are added to the overhanging features to enable support-free printing. For the steel and wood voxels, the faces are split into halves to enable better packing on sheet stock, as shown in Figure 5-3. With this 2D decomposition, we are able to achieve a peak continuum packing density of just under 45%. As this still represents significant material waste, a potentially interesting avenue for future work would be to consider optimization methods for adjusting the voxel decomposition to facilitate improved nesting density towards lower embodied carbon values.

Robotic Assembly System

To make more accurate projections about the assembly speed, cost, and energy consumption of a voxel-based construction approach, we developed a new mobile robotic assembler system. Our robot is based on the inchworm-style paradigm used in [89], [161], and [71]. Our system, called the Modular Inchworm Lattice Assembler robot, or MILAbot, is a five-degree-of-freedom robot that traverses a voxel structure using grippers on both ends, and can deliver and place a voxel payload using three additional voxel grippers. The MILAbot is shown in Figure 5-4 for reference.

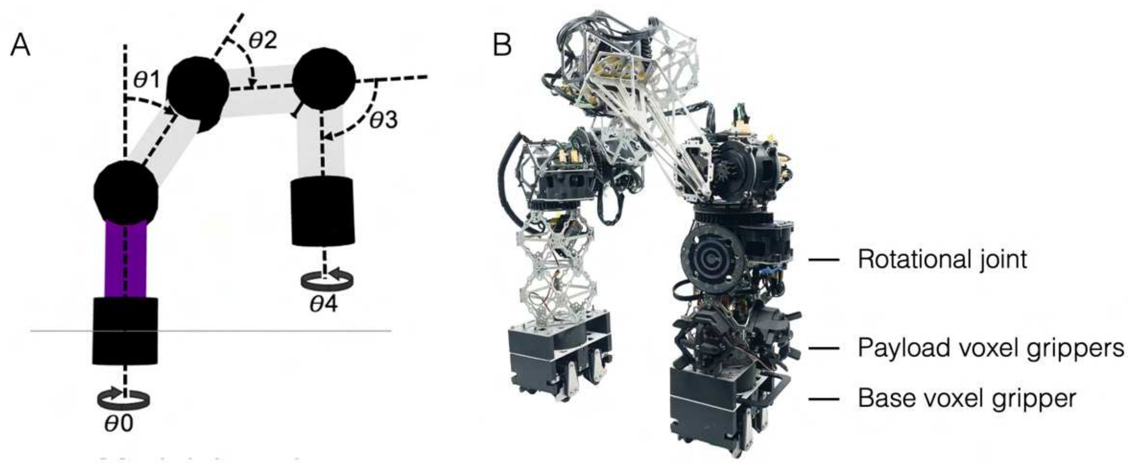

Figure 5-4: Inverse kinematics model and physical hardware of the modular inchworm lattice assembler robot (or MILAbot).

The MILAbot operates by carrying larger compounded blocks of voxels and adding them to the structure by dropping the blocks and engaging the snap-fit connection. Larger structures are built up by staggering the blocks such that the entire structure is fully connected, while the protruding octet, snap-connection, and dovetails provide stability under bending and tensile loads, as discussed in the prior section. By building with pre-offset blocks, the MILAbot is able to assemble first layers and overhangs. The intended workflow of this system is that voxels are pre-compounded into larger blocks via an automated system and then delivered to site where they are then assembled into larger structures by the MILAbots. The development of an automated system for this pre-compounding is beyond the scope of this work, but because this would occur in a controlled environment for relatively small structures, we anticipate that it would be possible to leverage existing high DoF gantry or robotic arm systems, which have been used to produce structures of similar or higher complexity [36], [177], [169] in a similar manner to pre-production proposed in [172] or [135].

The MILAbot is controlled through a Web-based real-time digital twin simulation tool, as shown in Figure 5-5A. Structures may be either imported as mesh geometries and then voxelized, or may be designed by hand with a point and click interface. From there, robots and voxel feeds are placed, and the structure is partitioned between the robots, and an assembly sequence is determined. Partitioning is based on grouping the voxels based on distance to a voxel feed, such that there is no overlap between different robot domains, which innately prevents collisions. Once an assembly sequence is determined, robot paths are determined based on a modified A* algorithm as described in [89] as this provides good performance even over large structures. At each step, commands are streamed from the digital twin to the physical robots over Wi-Fi. A more complete technical description of the implementation and validation of this system is available in [163].

An example assembly sequence is shown in Figure 5-5. First, voxel blocks are manually

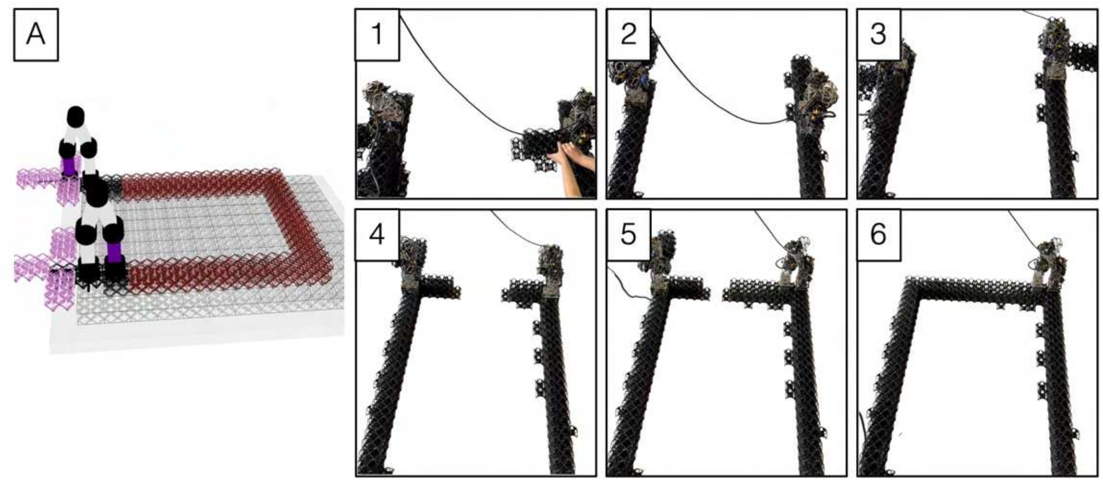

Figure 5-5: Overview of the robotic assembly of a doorframe, showing the digital twin control space and key frames of the assembly sequence.

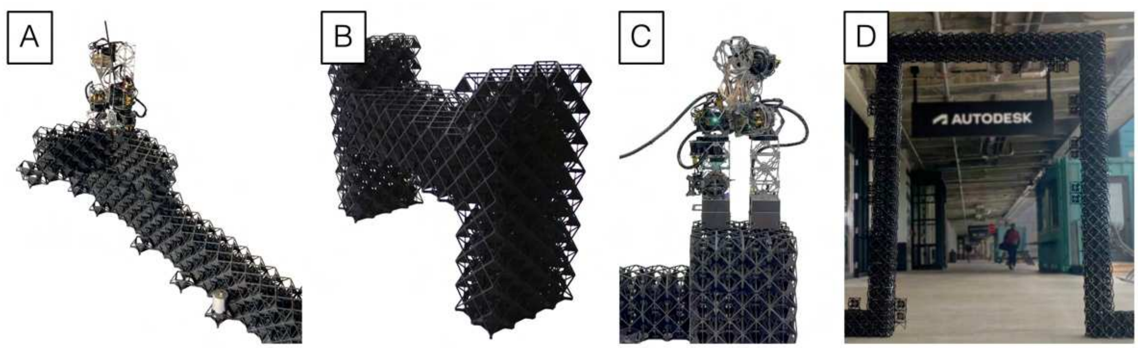

Figure 5-6: Example robotically-assembled meter-scale voxel builds.

fed to each robot. Then, the robots step forward and iteratively place new voxel blocks, returning to the feed to pick up new blocks. In this sequence, the robots assemble a doorframe flat on the ground, which we then tilt up, as shown in Figure 5-6. Note that the fringes that are visible on the built doorframe are the result of using compounded voxel blocks with extra voxels added on the sides to act as lateral stabilizers, as this assembly demo was done on top of a cloth backdrop that we could not rigidly fix the voxel feeds or any of the first layer voxels to. In a "real" first layer build, this would not be necessary, as we would anticipate that the first layer of voxels would be anchored a foundation.

We used the MILAbots to assemble a set of meter scale objects to determine the assembly throughput of the system, as well as to better understand its limitations. The assembly demos are shown in Figure 5-6. Based on these, we determined the average time per movement of the robot as 9.5 seconds and the average power consumption as 13.3 W. We additionally identified some key limitations of the hardware.

The first significant limitation is that the robot is unable to climb vertically, which means that it needs to build a support staircase or to assemble horizontally and then tilt up, as in the case of the doorframe. This limits its ability to scale to larger structures, as the footprint and assembly time for the stairs also grow increasingly large. So, we proposed that a more scalable system might use a vertically actuated voxel feed, which engages with the voxel structure equivalent to a pinion gear on a rack, similar to what is proposed in [44]. Although this is not yet implemented in hardware, we implemented it for the simulation software to extrapolate assembly time estimates.

Next, our system currently uses manually fed voxels, which is also a significant limitation on the throughput of the system. This will also need to be automated if the system were applied to the architectural scale. The goal would be to have a flat-packed stock of parts that would be assembled on site by a desktop-scale CNC, with the assembled voxels then distributed to the MILAbots via an automated system.

Lastly, the actuators in the robot use 3-D printed gearboxes that fatigued and eventually failed during our testing. For the purposes of making a more accurate cost estimate, we replace all of these components with an equivalent metal off-the-shelf reduction. This takes the total cost of a single robot from $750 to $1,250. To account for the cost of an elevator system, we then increase the estimated per-robot cost to $1,875.

Structural Element Design

After determining the relevant parameters of the voxels and robotic assembly system, we designed the overall voxel building. The simplified building consists of a fully enclosed 5 x 10 x 3 m box with approx. 610 mm spacing between framing elements. The structure is sheathed internally and externally with 1/2" oriented strand board (OSB) and a 100 mm thick layer of expanded styrene foam (EPS) insulation. The wall studs have a square cross-section formed by voxels, while the roof joists have an I-beam cross-section formed by voxels. This is shown in Figure 5-7.

We determine the required quantity of voxels for the structure by treating each 5 m-long 3 m-tall spanning element (consisting of a stud on either side with a joist over the top) as an individual structural element that needs to support its own self weight, that of the sheathing and insulation on it, and an additional 20 p.s.f live roof load based on the requirements in the ASCE 7 [24]. We then increase the dimensions of the framing element until the maximum displacement is below the L/360 standard [82]. This is summarized in Figure 5-8.

To evaluate the structural performance of each archway (the stud-joist-stud spanning element shown in Figure 5-8), we performed Finite Element Analysis (FEA) to determine the maximum deflection under a consistent load case. To achieve this, we developed a simulation framework using Python-Abaqus scripting. The script generates the arch geometry based on the specified stud and joist cross-sectional properties. We treat the mechanical properties of the voxels as the properties of the bulk material in each archway. Depending

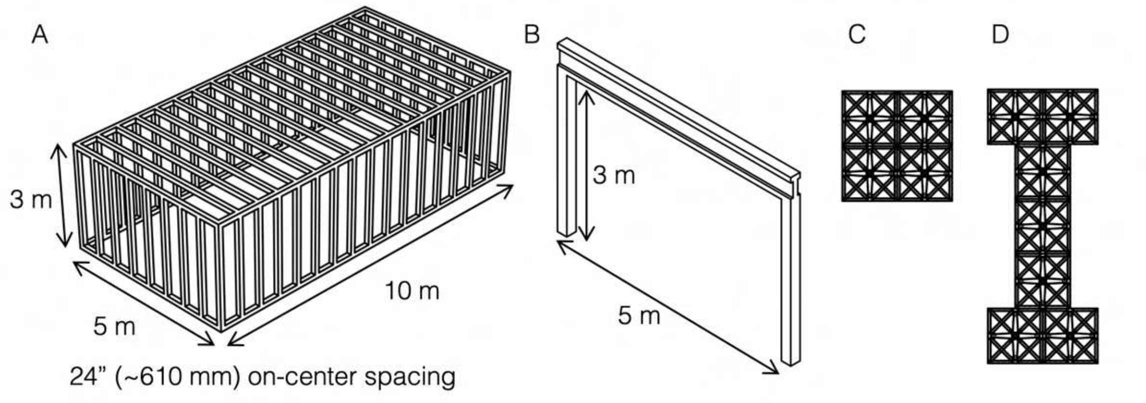

Figure 5-7: Diagram of the simplified structure used for our estimates, showing the base plan for the 24" (610 mm) on-center spaced stud framed 5 m x 10 m x 3 m building, the single spanning frame element used for evaluating the voxel approaches, and example cross sections of the stud and joist elements.

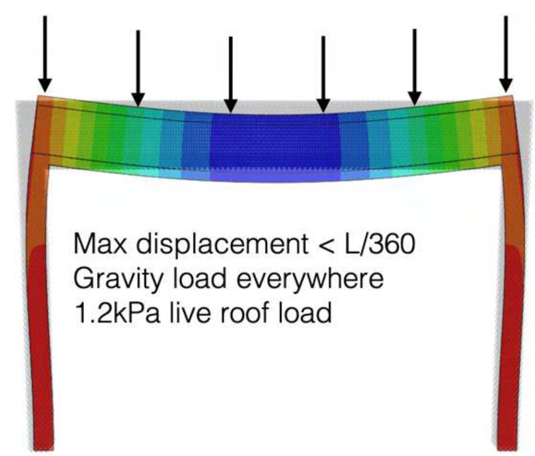

Figure 5-8: Overview of the evaluated loading scenario, with a 30x deformed model overlaid on top of the undeformed structure.

on the voxel type and the availability of the data, the Young's modulus is typically derived from compressive modulus measurements of either single unit cells or assemblies up to 4×4×4 voxels. Density values all account for the mass of connectors in addition to the voxels themselves. The script then defines the boundary conditions: the bases of the beams are fixed, a gravity load is applied, and a pressure load is imposed on the top surface. Due to the geometric variability of the models, the pressure load is calculated such that the resulting equivalent point load is 3.66 kN. The generated solid is then meshed using C3D10 tetrahedral elements. We chose this element because it is easy to automate while offering 10 nodes, quadratic interpolation, and four Gaussian integration points, ensuring good accuracy and a proper representation of the field variations across the element. The mesh size is half of the voxel height value. Finally, the script runs non-linear analysis and, once converged, records the maximum deflection of the horizontal beam in the vertical direction. This code is provided in 5.9.

While treating the voxel mechanical properties as equivalent to that of a bulk material is a simplification, prior work indicates that the voxel behaviors extrapolate well: one of the key objectives behind each voxel design is for the geometry of the lattice to dominate over that of the joints and behave as a bulk material with equivalent properties. Prior work has established that joint effects are second order compared to that of the beams [91]. In line with this, discrete systems achieve similar performance to their continuously manufactured counterparts. For example, the discrete lattices in [46] exceed the performance of the continuous microlattice described in [157], while the voxel system in [72] achieves comparable performance to the continuously manufactured lattices in [184], indicating that behavior of the beams dominates.

To better validate our modeling, we set up four samples of voxel beams to test under 3 point bending as a physical and virtual experiment. These results indicate strong alignment between the physical discrete assemblies and their virtually modeled continuum material counterparts. The full results and description of this are available in 5.9. However, it is important to note that our results here are based on a first order approximation of the actual system, and that real deployment of any of these systems would require more physical testing as well as more system-specific validation of FEA tools and workflows.

In larger assemblies, manufacturing tolerances and cumulative geometric slack can introduce localized nonlinearities that deviate from idealized bulk behavior. The voxel systems considered here are designed with geometrically constraining node interfaces to minimize slip accumulation and error propagation across scales. Although prior experimental studies have shown that stiffness remains beam-dominated rather than joint-dominated in relatively large assemblies [46], [51], tolerance accumulation may become more pronounced in structures at the building-scale. While the modeling framework we use reflects the intended mechanical regime of these systems, it should be noted that it is an idealized approximation of the system.

A summary of all of the voxel beams, along with their maximum deflections, are shown in Figure 5-9. We adjusted the cross-sectional voxel quantities until the deflections were within 2 mm of each other, and all below the maximum of 13.89 mm. As Figure 5-10

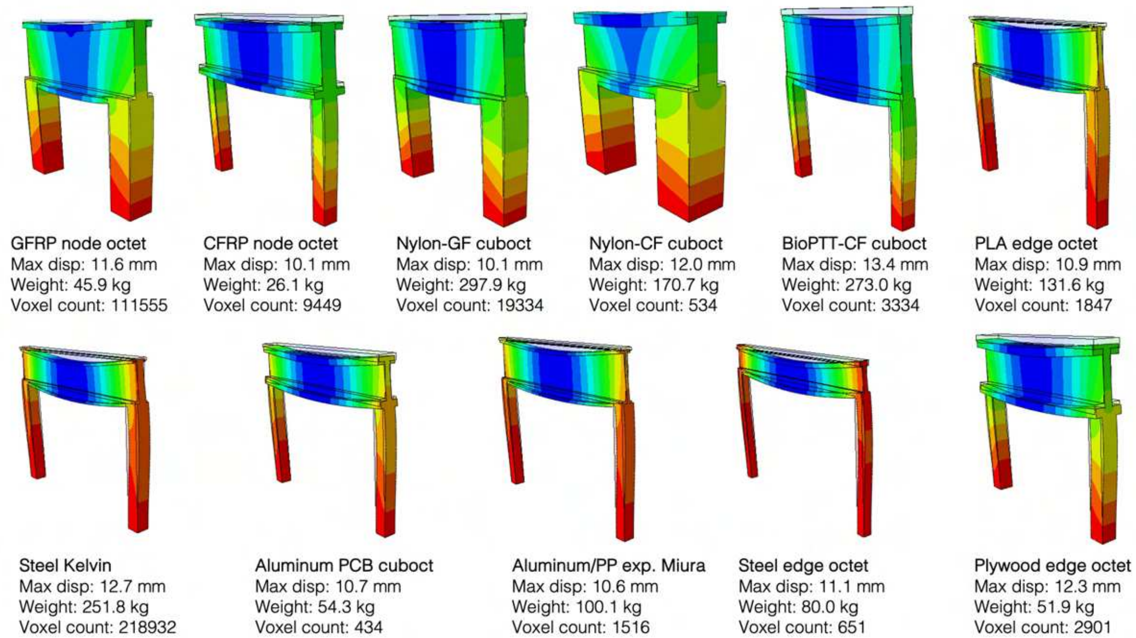

Figure 5-9: All of the evaluated voxel frames, with the maximum displacement, single frame weight, and single frame voxel count noted.

shows, the dimensions, voxel quantities, and framing weights vary significantly across the different voxel approaches. This is because the voxels span a range of mechanical properties, so for an equivalent absolute load on the roof, some voxel approaches require substantial dimensions before they reach the requisite stiffness to support it. The total weight of the framing structure is then the weight of all of the load-bearing component frames, along with the short-side stud elements (square cross-section), as well as the upper and lower shear bracing (I-beam cross-section), as shown in Figure 5-7A.

We projected the assembly speed for each of the frames by determining the average number of actions per placed voxel for a single structural framing element using an elevator-style voxel feed. "Actions" are defined here as each robot step type and each voxel placement type, which are structured as two-move sequences to return the robot to the same starting configuration, with the legs placed two voxels in x or y away from each other. For example, the single action of "step forward" consists of moving the relative front leg forward to a spacing of four voxels horizontally away, and then moving the back leg to close the gap. Voxel placement movements are similarly structured, so a placement movement looks like swinging the back leg into the placement position, placing the voxel, and retracting back to the default position. Essentially, the library of actions consists of all steps that can move the robot two-voxel distances laterally, up to two-voxel distances laterally and vertically, or place a new voxel block within that bounding envelope. Because of the relatively similar distances traversed during each of these movements, and because of the reliance on mechanical alignment features over precise robot control, all of these steps take roughly a similar amount of time, allowing us to extrapolate approximate robot throughput speeds based on

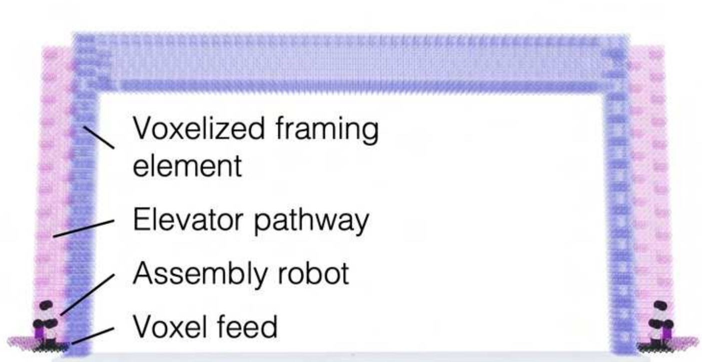

Figure 5-10: One of the structural elements loaded into the MILAbot simulation space, configured for a moving voxel feed.

the amount of actions taken during a build.

Using this sample framing element build, this results in an average of 6.5 actions per placed voxel. At 9.5 seconds per movement, carrying the maximum volumetric payload of three 4x2x2 65 mm lattice pitch voxel blocks, this results in a volumetric throughput of 0.26 m^3 per hour. For our projections, we use 25% of this value to build in sufficient buffer to account for set-up and de-installation time, as well as a future voxel feed system.

5.4.2 Compared Construction Approaches

Here, we compare voxel-based approaches with 3D concrete printing (the predominant onsite automated method), precast modular concrete (a prefabricated method), concrete masonry units, cold-formed steel framing, and stick framing (manual methods). Material amounts and embodied carbon estimates are drawn from a range of sources in the literature, while overall material, labor, and equipment costs are estimated using RSMeans [151] for United States national averages.

3-D Concrete Printing

For the 3DCP approach, we use a 40 mm track width and a wall cross-section similar to the ones in [30],[175], and [126]. The dimensions are shown in Figure 5-11. Based on commercial practices, such as what is discussed in [34], we additionally reinforce the wall by casting concrete with reinforcement into the walls according to the ratios given by [126], and adding spray foam insulation to an equivalent thermal insulation rating as the EPS foam used in

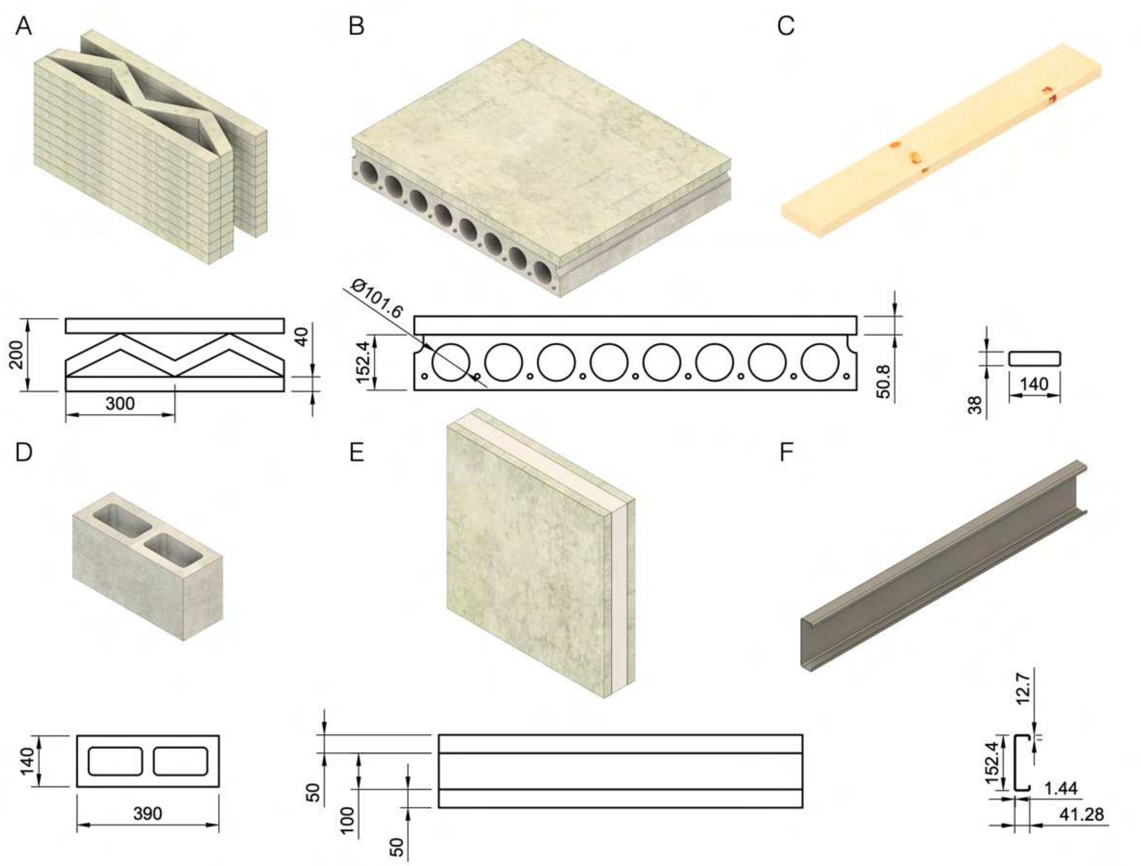

Figure 5-11: Summary of key parameters of the existing construction techniques that we compared against. All dimensions in mm.

the voxel and other framing approaches. For the roof, we use hollow-core panels, as in [30]. The dimensions of these are shown in Figure 5-11, and further discussed in the following subsection. We use the parameters of the COBOD BOD2 gantry girder printer to determine the build time. We determine the embodied CO_2e by using the values determined by [126] adjusted for our geometry. From this, we get an embodied carbon value of 17338 kg CO_2e, at a total weight of 46,267 kg. Using the BOD2 maximum printing parameters [49], the total time to print the walls is only 10.4 hours. Based on the BOD2 recommendations, we estimate another 10 hours for total set-up and de-installation time, with four printer techs on site at $30/hour for the entire print use time. The addition of concrete reinforcement, insulation, and pre-cast roof panels then adds another approximate 100 hours of build time, for a total assembly time of 119.7 hours. The overall material cost for the entire structure is $12,972.95, while the combined labor and equipment cost is $7,616.13. To estimate the cost of using the printer, we assume a five-year 5000-hour/year use schedule of the printer that we amortize its base cost ($600,000) over, in addition to the cost of the electricity needed to run it, using the 2023 Massachusetts average rate of $0.34 per kilowatt hour, similar to the method used in [175]. This results in a total cost $20,589.08 for the 3DCP building.

Precast Modular Concrete

For the precast modular concrete approach, we use insulated sandwich panels for the walls, [134] , and hollow core panels for the roof, as done in the 3DCP case. These are chosen according to [83] to meet the loading requirements and spans of the sample building. The insulated sandwich panel, as shown in Figure 5-11 consists of two 50 mm thick reinforced concrete slabs around a 100 mm thick EPS foam insulating layer. The concrete slabs are reinforced at a ratio of 33.5 kg/m^3, lower than the reinforcement ratios provided by [30], and more in line with guidelines given by [83]. The roof panels are then made of a 6" (152 mm) tall hollow core panel with a 2" mesh reinforced concrete topping. The cross-sectional area of the hollow core panel alone per 4-foot (1.2 m) section is 0.12 m^2 and has 9 strands of 9.42 mm diameter steel reinforcement. The topping is reinforced at 0.25% of its cross-sectional area. We assume all the concrete in the structure is the same blend, given by what is used in [126] . The total weight of this structure is then 42114 kg, with an embodied carbon of 11899 kgCO_2e. The total material cost is $23,681.24, while the fabrication cost is $2,676.99, for a total cost of $26,358.23. The assembly time is 54.25 hours. For this approach, and the remaining manual approaches, the cost and time estimates are generated using only RSMeans data, as previously discussed.

Concrete Masonry Units

The concrete masonry unit wall is designed according to reinforcement ratios given by [126] and adjusted for our dimensions. Like [67], we use 140 x 390 x 190 mm CMUs. We use the same hollow core panels for the roof as in the prior two concrete building examples. We also add a single layer of EPS foam insulation and OSB sheathing internally to the structure. The total weight of the structure is 48022 kg, and the embodied carbon is 14005 kg CO_2e.

The material cost is \$14,250.24, and the labor and equipment cost is \$9,822.80 for a total cost of \$24,073.04. The total build time is 209.03 hours.

Cold-Formed Steel Framing

The cold-formed steel framing follows the same basic pattern as the voxel framing scenarios, though we use a constant cross-section throughout the structure. The dimensions of this are shown in Figure 5-11 and are taken from a product listing for a 16-gauge 6" stud [48]. For the roof joists, we place these back-to-back for added stability and verify that the structure deformation is below the L/360 standard under the same loading conditions as used for the voxel approaches. Sheathing is then added as in the voxel approaches, for a total structure weight of 2074 kg. The embodied carbon is 6608 kgCO_2e. The material and labor costs are \$15,859.03 and \$10,998.60, respectively, for a total cost of \$26,857.63. The build time is 282.1 hours, for two carpenters working on each task sequentially.

Stick Framing

As with the other framing-based approaches, the stick framing case follows the same stud pattern. We use 2"x6" (real dimensions 38 mm x 140 mm) dimensional lumber as the base unit for the studs, and 2"x12" (38 mm x 280 mm) beams for the roof joists. We use steel ties to secure the roof joists to the studs. Including sheathing and insulation, the overall weight is 1149.85 kg and the embodied carbon is 1933 kgCO_2e. The material cost is \$4,983.09, the labor cost is \$4,981.26, and the total cost is \$9,964.35. The build time is 108.5 hours, for two carpenters working on each task sequentially.

5.4.3 Comparison Methodology

All of the approaches are compared using results for the entire simplified building, unless otherwise noted. This means that all the calculations include the additional finishes (sheathing and insulation), so that the comparisons are at an equivalent level of finish. For example, for the voxel approaches, this means that the time calculation is based on the robots first assembling the framing structure, and later a team of two carpenters insulating and sheathing the building sequentially. Similarly, cost accounts for price of the voxels, as produced by a vendor, amortized material cost and electricity costs of the robots, labor costs to supervise the robot build, and the material and labor costs for the traditional construction activities. For cost data from RSmeans, the base cost is taken to be comparable to how voxel costs were determined.

The embodied carbon assessment follows the principles of ISO 14040/14044 [86],[87], and is intended as a comparative assessment of alternative structural systems, as opposed to a complete cradle-to-grave LCA. The system boundary is limited to cradle-to-gate material production, corresponding to A1-A3 in EN 15804, encompassing raw material extraction, processing, and manufacturing. Construction stage activities, use phase impacts, and end-of-life scenarios are excluded, with the exception of the automated methods (3DCP and voxel approaches), where we factor in the electricity cost of running the automated systems in the overall calculation, as we consider this a necessary part of producing the material. For biogenic materials, we use the figures from [173], which covers A1-A3 modules, and excludes any biogenic carbon flows. Material quantities are derived from the previously described structural designs, including all additional sheathing, insulation, and fastening beyond the structural elements. For the voxel based approaches that leverage subtractive methods, i.e. the plywood and steel voxels developed for this work, the aluminum PCB voxels from [162], the steel Kelvin voxels from [95], and the expanded Miura plate lattice from [139], we include the off-cuts in our overall material consumption, as this represents a potentially non-trivial amount of material waste in the process. Under these assumptions, the embodied carbon results presented here are intended to support relative comparison of structural strategies and construction paradigms, rather than to provide absolute environmental impact values for specific buildings.

5.5 Results and Discussion

In this section, we compare and evaluate the voxel approaches against the existing construction approaches in terms of their sustainability, cost, and building speed. These criteria reflect primary environmental, economic, and logistical drivers in construction decision-making. While other impact factors are relevant to the environmental sustainability of a construction approach, we use embodied CO_2e, as this metric is widely reported and allows consistent comparison across systems. A complete summary of these parameters is also available in Table 5.3.

Sustainability Assessment

We use the embodied carbon of each approach as our metric to evaluate its sustainability. While the embodied carbon of many of the voxel approaches is quite high, as summarized in Table 5.3 the three voxel types developed for this work demonstrate potential sustainability benefits compared to traditional construction methods. As shown in Figure 5-12, the voxel-based approaches outperform all of the concrete-based methods, while the steel and wood voxels are able to provide lower embodied carbon than every approach, aside from stick framing. Compared to 3DCP, the steel voxel approach generates only 36% the amount of the embodied carbon, while the plywood approach generates only 17% of the embodied carbon of the 3DCP approach. The emissions of the voxel approaches are more comparable with their equivalent framing-based approaches, but achieved via a potentially automated building process.

However, most of the considered voxel approaches perform poorly in terms of sustainability. The voxel-only contribution to the embodied carbon is plotted for all voxel types in Figure

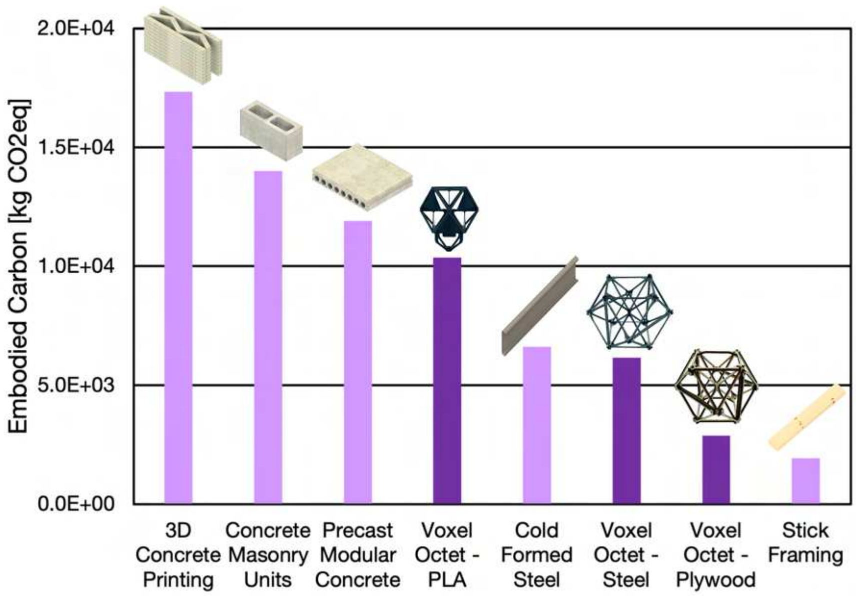

Figure 5-12: Total embodied carbon for the three voxel approaches developed for this work, compared to the standard construction approaches for the complete simplified 1-story building.

5-13. Of the existing construction approaches, 3DCP had the highest embodied carbon value at 17,338 kgCO_2e— this value is exceeded by the voxel contribution alone for all the voxel approaches with higher embodied carbon than the GFRP node-octet.

This wide range is because while some voxels were designed for improved sustainability, most were designed only for specific mechanical performance metrics. To compare the material-sustainability efficiency of the voxels we can evaluate their specific modulus (MPa/ρ) relative to their carbon intensity (kgCO_2e/kg). This metric helps capture the balance between the mechanical efficiency of the lattice (how stiff it is relative to its density) with its environmental efficiency (how much embodied carbon is in each unit of the material system). So, for example, although the aluminum cuboct has relatively good specific modulus at 0.78 MPa/kg/m3, this is outweighed by its very high carbon intensity of 30.17 kgCO_2e/kg.

From this metric, we see that the plywood and steel octets are significantly more efficient than the other considered approaches. This is because they are manufactured from base materials with lower embodied emissions compared to the other approaches (especially the aluminum and fiber-reinforced high-performance plastics) and produced in a way that does not waste too much of the feedstock (in contrast to the steel Kelvin lattice, which uses subtractive fabrication without considering packing efficiency). This is additionally because the chosen lattice type, the edge-connected octet, has better mechanical efficiency at the ranges needed for building loads. From this metric, the next most efficient systems are the PLA edge octet developed for this work and the CFRP node octet, which could have potential improvements through substituting their constituent materials with more sustainably efficient materials.

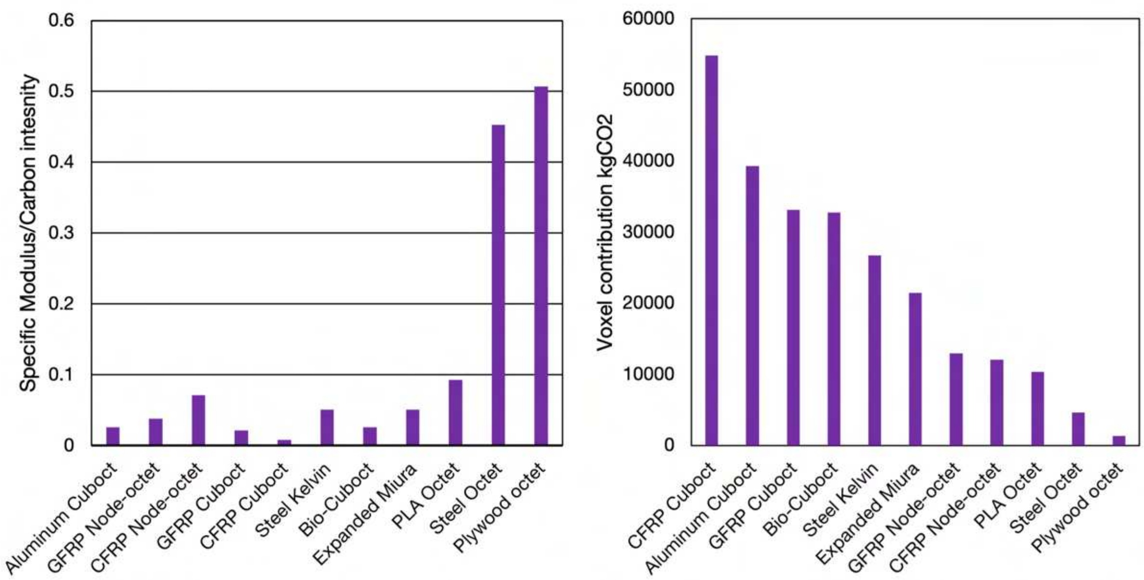

Figure 5-13: Embodied carbon [kgCO_2e/kg] of the voxel approaches relative to the voxel's specific compressive modulus [MPa/ρ] and embodied carbon of each voxel building approach, accounting only for the voxel contribution.

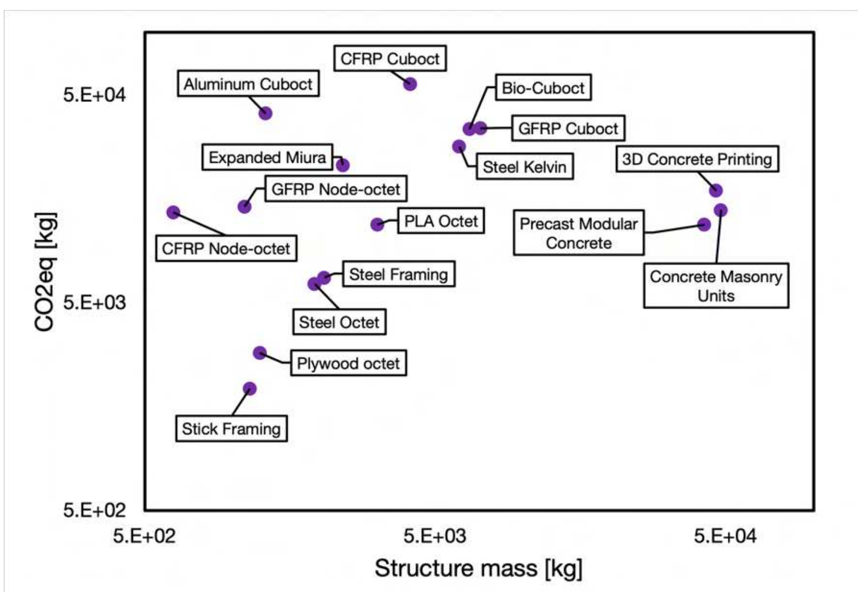

Figure 5-14: Overall structure mass, including sheathing and insulation, compared to embodied CO_2e for all of the considered approaches.

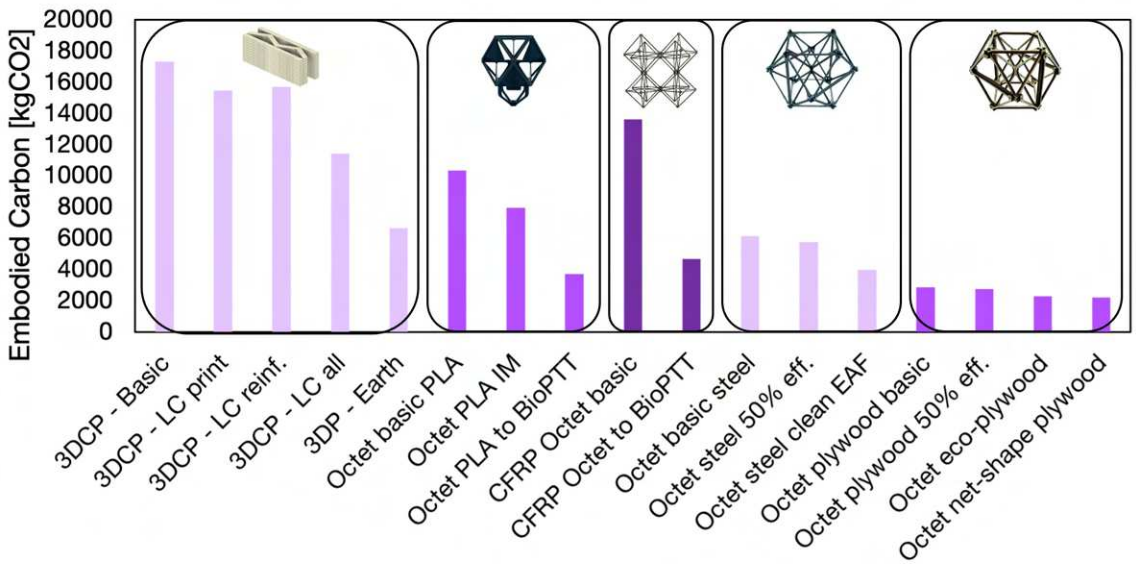

Figure 5-15: Projections of the sustainability performance of 3D printing and voxel approaches with material substitutions.

We can also evaluate the overall material sustainability of all the approaches by considering the ratio between their embodied CO_2e and their overall structural mass. These results are plotted in Figure 5-14. As this graph shows, the concrete approaches have lower embodied carbon per mass than any of the other approaches. However, in absolute terms, they still have an overall high embodied carbon. This indicates that their building approaches are inefficient in terms of how mass is distributed, suggesting that work towards more efficient concrete usage could offer significant sustainability benefits. In contrast, the performance of the CFRP node-octet, for example, is already extremely lightweight, but with very high emissions, indicating that the most promising route for improving performance would be through base material substitution.

Based on these results, we additionally consider some projections for the different building approaches with more eco-friendly material substitutions. These are summarized in Figure 5-15. For the 3DCP approach, we consider the impact of switching to a lower carbon concrete (denoted as LC in the graph) for the 3D printed portions, the cast-in-concrete reinforcement, and both. For these calculations, we use the figures reported in [128]. By substituting lower embodied carbon concrete for all of the concrete, we are able to reduce the embodied carbon of the 3DCP approach from 17,338 kgCO_2e to 11,441 kgCO_2e, a 66% reduction. There is also substantial work exploring alternatives to concrete, such as printing using earthen materials, as by WASP [12]. Using the reported carbon intensities from [52], we find that substituting an earthen mixture with a low-carbon intensity, the 3D printing method can further lower to 6,655 kgCO_2e, which is competitive with our projections for the steel octet voxel.

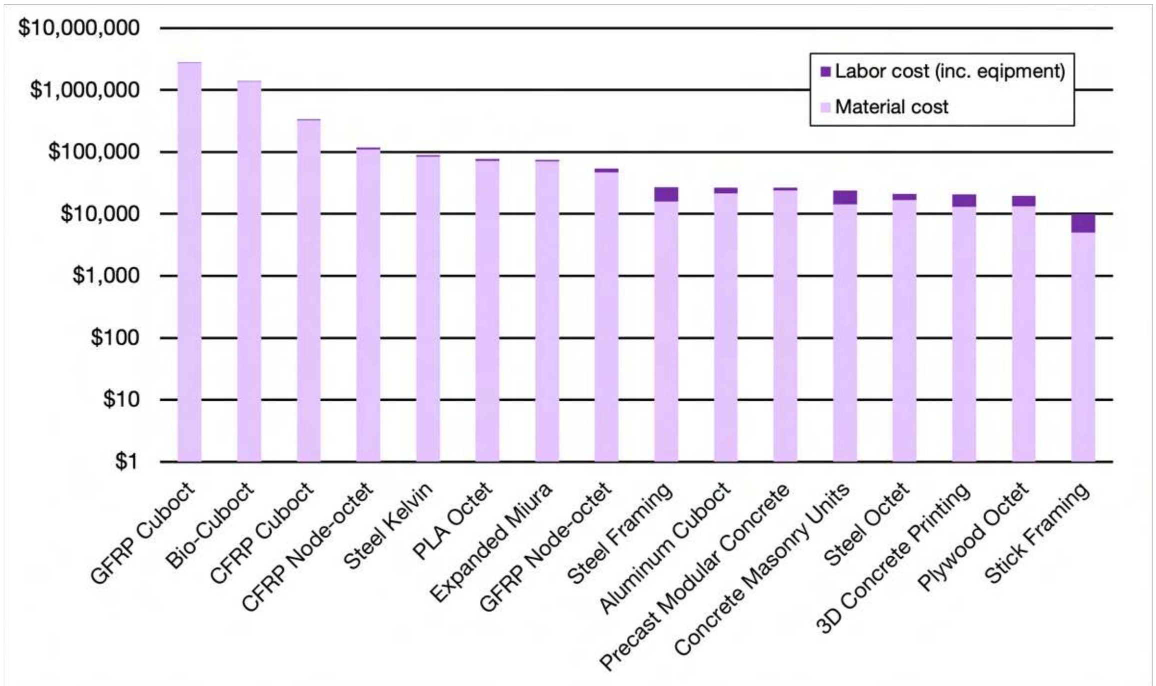

Figure 5-16: Material cost, consisting of the purchased material cost of all components, and fabrication cost, consisting of the labor cost and equipment cost for each approach.

Building Costs and Times

Cost data is determined as previously described. Note that for assembly times and labor cost calculations, for the voxel approaches, we assume a robotic swarm of 24 robots operating symmetrically over the structure, with a volumetric assembly speed that is 25% of the speed of the current hardware system, to build in a buffer for system installation/de-installation and voxel feeding. Sheathing and insulation are then manually applied sequentially after the structural assembly.

The voxel approaches span a wide range of costs; however, most of the approaches are within the same order of magnitude as the existing construction approaches. The steel and wood octet voxels slightly outperform most of the considered existing approaches. The cost of every considered approach is shown in Figure 5-16. Figure 5-17 shows the total assembly time for the traditional construction approaches, as well as the considered voxel approaches. Steel framing is the slowest approach, at 282 hours, while precast modular concrete is the fastest at 54.3 hours. The voxel approaches fall in a similar range as the stick framing and 3D concrete printing approach.

The stick framing-based approach is by far the cheapest approach in total, while the steel-framed approach is the most expensive of the existing construction approaches. The total cost of both framing approaches is approximately evenly split between labor and material costs, representing the highest proportional contribution by labor of all the considered

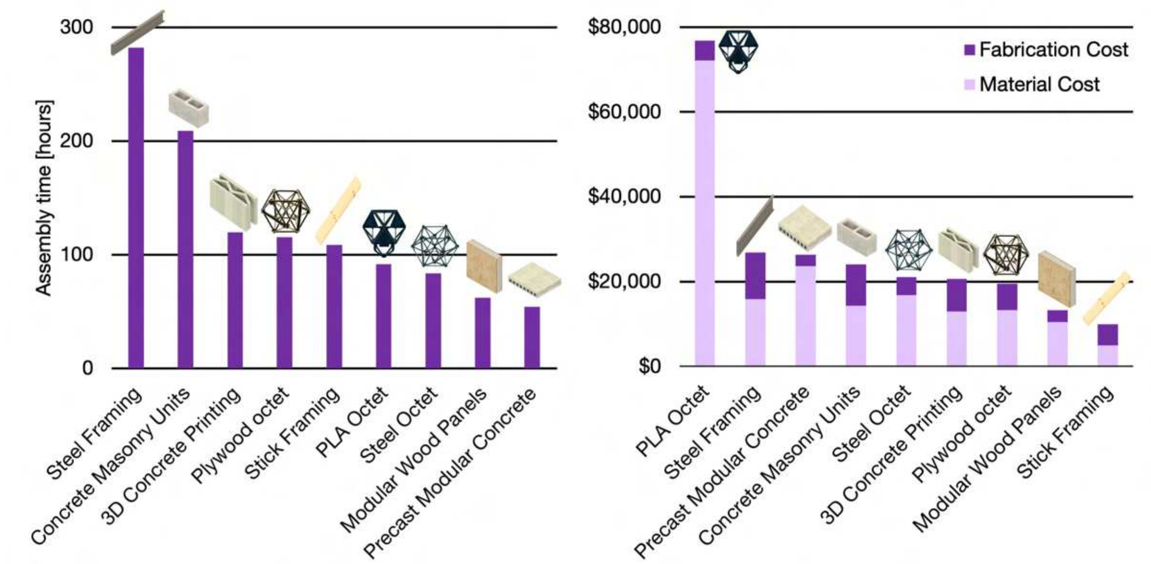

Figure 5-17: Total assembly time for each approach and total construction costs of the existing construction approaches and only the voxels developed for this work.

approaches. On the other side, the pre-cast modular concrete approach has the lowest labor contribution of the considered approaches.

Given the low labor costs pre-cast modular concrete and the low material costs of stick framing, it is worth considering the performance of other wood-based modular building systems. Structural insulated panels (SIPs) consisting of two OSB layers around an EPS foam interior sandwich-style panel provide a wood-based alternate construction system we can consider in relation to the voxel approaches. Projections of the speed and cost of a SIPs version of the building, based on the same RSmeans dataset, are shown in Figure 5-17, along with the other existing construction approaches and the voxel approaches developed for this work.

Compared to 3D concrete printing, the voxel assembly method achieves relatively similar speeds for the structural components. The 3DCP approach only requires 20.4 hours to print the walls, in addition to installing and deinstalling the printer, according to figures from COBOD [49]. The steel voxel requires 7.1 hours, the PLA voxel requires 15.2 hours, and the plywood requires 39.1 hours, including setup and de-installation time. A significant portion of the total build time for both automated approaches is spent on sheathing and insulation (about an additional 80 hours for both approaches), highlighting that efforts aimed only at automating the construction of the base structural system may be limited in the total speed improvements that they can achieve. This is emphasized by the fastest speeds achieved by the modular wood panels (SIPs) and the precast modular concrete panels, as in both of these approaches, there are no secondary passes for insulation and sheathing.

This suggests that while the robotic assembly system could achieve further throughput improvements through using higher-performance motors, increasing the swarm assembly

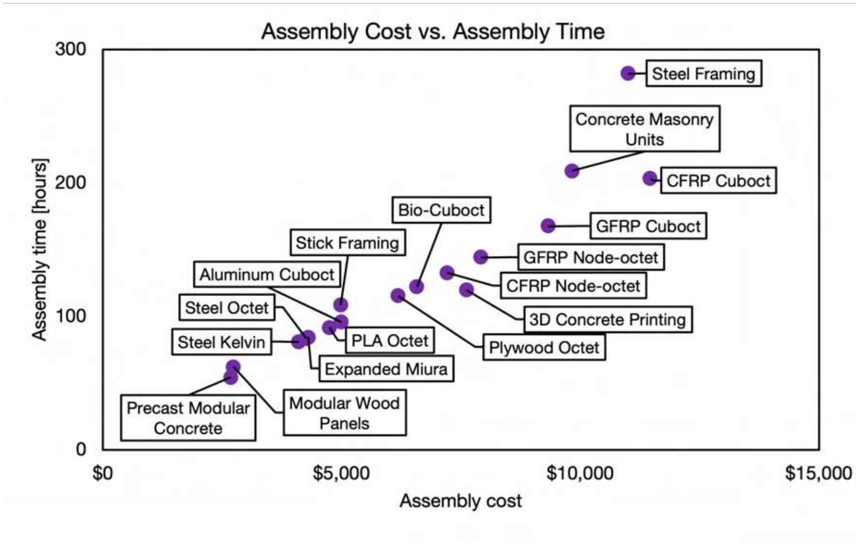

Figure 5-18: Assembly cost versus the overall assembly time for every considered approach.

size, or increasing its carrying capacity, the greatest potential benefits lie in additionally automating future operations, such as sheathing and insulation. The approach for this could either be sequential, that is, the mobile robots install the sheathing as a second pass, or it could be done in parallel, by developing a closed, more brick-like voxel geometry. A future system incorporating sheathing-integrated voxels might also have insulation built in, and use electronics-integrated voxels sparsely to additionally build out the wiring of the house.

In Figure 5-18, we plot the assembly cost versus the assembly time, showing a relatively linear relationship between the two. This is because our cost model assumes two workers supervising the robotic voxel assembly system, as well as manually installed sheathing and insulation, so the overall fabrication cost is dominated by labor as opposed to the cost of using the robot, which is low: less than $20 for all voxel approaches, and less than $10 for the three voxel approaches developed for this work.

The amount of robots to use for a given structure represents an interesting problem. While initially throughput can be trivially increased by increasing the number of robots, this will reach a limit either physically, based on building footprint or structure shape, or based on the challenges of computationally planning for very large swarms, or based on the logistics of managing a large group of robots in hardware. This is further discussed in the following subsection. For this structure, we use 24 robots because this amount results in both full utilization of the swarm based on a simple partitioning of the structure into roughly equal voxel-count substructures, while also balancing against logistics challenges of handling larger robot swarms, which is further discussed in the following subsection.

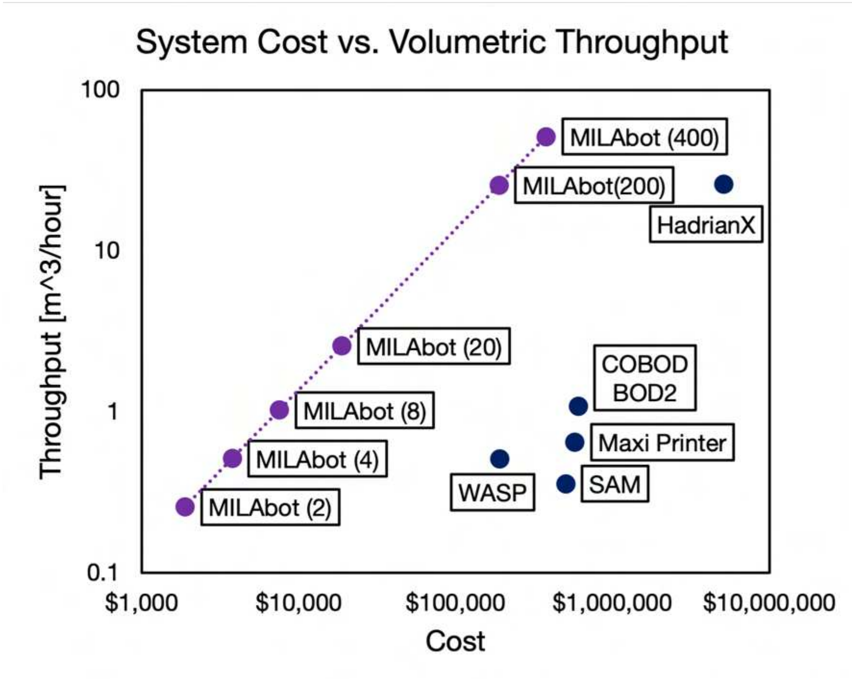

Figure 5-19: System cost and volumetric assembly throughput scaling for increasing MILAbot swarm sizes as compared to selected existing automated concrete printing and bricklaying systems.

Automated System Throughput

The robotic system for the voxels is focused on being cheap and simple, and relies on parallelization to achieve improved throughput. This stands in contrast to existing commercial approaches to automated structural building, which typically rely on expensive and large machines. In Figure 5-19 we plot the volumetric throughput of the MILAbot system against some other existing systems: the WASP mud printer (gantry system) [12], the SAM bricklayer [11] (robot arm on a linear base), the Maxi concrete printer [14] (robot arm on a mobile base), the BOD2 concrete printer [49] (gantry system), and the Hadrian X bricklayer [13] (robotic arm on a crane with a mobile base). These prices are taken from the starting prices listed on each of these system's respective sites, while the MILAbot cost is doubled from the production cost.

A single MILAbot is much slower than the other approaches; however, it is also much cheaper. By increasing the amount of MILAbots, we linearly increase the throughput, and we catch up to (or exceed) the BOD2, Maxi, SAM, and WASP systems at less than 20 robots. While a swarm size of 20 voxel assemblers is larger than what has been previously demonstrated, it is still in a clearly tractable range, especially for well-partitioned builds. The MILAbot system catches up to the throughput of the HadrianX system at 200 robots at an order of magnitude cheaper. However, such a large swarm size, especially for a relatively small project, poses significant logistics and planning challenges that would limit our potential ability to trivially scale to those robot counts.

As previously stated, while increasing the robot count can increase system throughput, at some point, this will run into geometric, logistical, or computational barriers that prevent continued improvement, or could even begin reducing the system throughput. When these barriers come into play is highly geometry dependent, as the shape of the structure, as well as the access point around it on a site, can create bottlenecks for robotic assembly. If we consider the example of a long, solid and thin wall, if we are able to place robots and voxel feeds evenly along the length of it, we can achieve better than linear improvements in assembly throughput [17]. If we are however only able to place robots and voxel feeds at one end of the wall, we will receive little to no benefit from adding further robots, as they will block each other along the path. Similarly, for roof structures or other overhangs, the quantity and size of their vertical supports dictates access for the robots. Although prior distributed robotic assembly work has demonstrated various strategies that can help mitigate this bottleneck effect, such as through bucket-brigade style behavior [65], our current path planning implementation does not accommodate this. Even if geometric constraints are not a limiting factor, the logistics of deploying large quantities of robots are relevant for real-world applications. That is, with a limited team of people to install and supervise the robots, increasing numbers of robots lead to increased installation and deinstallation times. While this may be a less pronounced problem for more high autonomy operation, such as in-space assembly tasks, for the tasks considered here, these times can dominate over the robotic performance. Finally, because for this scenario we can precompute assembly paths, while computational challenges can eventually become limiting for very large structures, we are unlikely to encounter these for this system before running into the previously described physical barriers to scalability. Various literature has also explored alternate control systems for more highly autonomous and dense swarm systems [73].

To better see how these geometric and logistical challenges impact arbitrary scaling, we can consider our example structure, as built using the plywood voxel system. We used our digital twin software to simulate different partitions of this structure and how throughput is impacted by increasing robot quantities for these substructures. Based on our prior assumption of two workers monitoring the robot build, we factor in installation time as proportional to the quantity of robots used, assuming that two robots can be set up at a time. Our prior calculations also assume that the robots and voxel feeds can been manually moved around the build site as substructures are completed, as this enables the robots to fully access the sample structure without temporary supports. The time this takes is factored in based on how many (if any at all) robots need to be moved around on to continue assembling further substructures.

The assembly throughput in m^3/hr versus the number of robots for the plywood voxel building is plotted in Figure 5-20, with a small parameter sweep of installation times and voxel feed switching times, to illustrate how these metrics impact overall throughput. The basic case is plotted in black, and assumes 0.75 hours for every two robots to be installed, and 0.15 hours for every two robots and voxel feeds to be moved to a new position. The alternate options vary only one of the parameters, as indicated by the legend. In the absence of the installation/deinstallation time considerations, throughput linearly increases until we

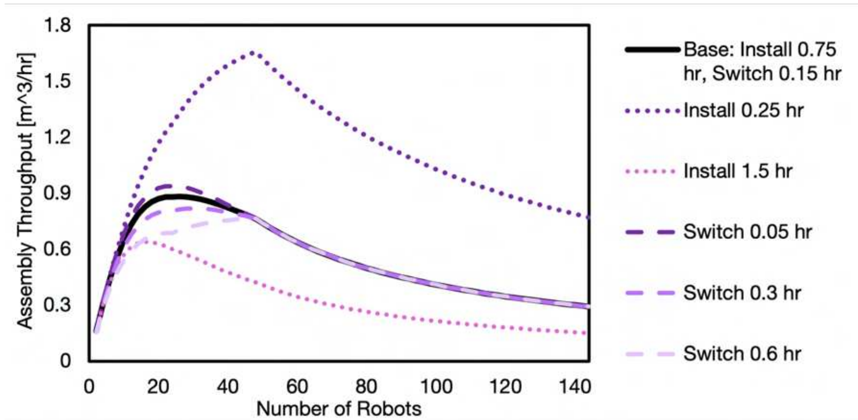

Figure 5-20: Assembly throughput in meters cubed per hour versus the number of robots for the plywood voxel based simplified building.

reach the maximum amount of robots that can build without sharing a vertical stud, which is 48. At this point, the throughput continues increasing at a much slower rate, as each stud has enough capacity for two robots to split the assembly, improving the overall speed. However, because of the narrowness of certain sections of the horizontal joist elements, the additional robots are not able to contribute to building these. The throughput continues increasing until we start increasing the robot count to three per stud, at which point they begin to interfere with each other, resulting in a decrease in the assembly throughput. The addition of installation and switching time considerations primarily serves to penalize larger quantities of robots, pushing the peak productivity toward lower robot counts. While the timings for these actions are based around our experience with the hardware, these are highly approximate and would likely require deployment outside of a lab environment to more realistically evaluate. However, these results help to illustrate how physical constraints around building geometry and likely necessary manual actions during the assembly process can limit overall scalability.

The comparison to existing systems in Figure 5-19 is intended to provide a simplified comparison point, as each of the considered systems face various limitations that impact both achievable assembly throughput and what types of geometry are buildable, making direct one-to-one comparison difficult. So, while Figure 5-20 clearly indicates that for a structure of this size, we cannot arbitrarily scale into the hundreds of robots, a structure with 10 times the wall perimeter (which does not mean it needs to be an overall 10 times larger structure), could much more easily support an increased number of robots. This suggests that a potentially fruitful avenue of future work could be to consider both which types of structures are best suited for distributed robotic assembly as well as how overall structure designs might be modified to better accommodate robotic fabrication.

5.6 Limitations and Future Work

We evaluate voxel systems within a complete building-level comparison framework that considers embodied carbon, total cost, and assembly time, unlike prior voxel and architected lattice work, which focus primarily on mechanical characterization [46], [184], [72] or system demonstration outside of construction contexts [90], [162]. While many prior robotic construction systems have been explored, these works often focus on demonstrating structural, robotic, or material performance [80], [113], [124] in isolation. Because our goal is to demonstrate the potential applicability of an entire material-robot system to building construction, a focus of this work is a normalized comparison of building approaches considering multiple metrics relevant to a construction system.

Similarly, much of the prior work on robotic construction emphasizes either structural performance or fabrication efficiency in isolation; here, we integrate structural simulation, robotic throughput modeling, and cradle-to-gate embodied carbon assessment under consistent boundary conditions. For comparability, all construction approaches, including additive and modular systems, are normalized to equivalent functional requirements and levels of finish, enabling a more consistent evaluation of automation strategies in realistic building scenarios.

However, while this work demonstrates the potential of robotic voxel assembly as an efficient and sustainable in-situ construction approach, there are still substantial limitations that need to be addressed before this approach is ready for practical deployment. The voxel approaches have only been demonstrated at a lab scale, so many important performance characteristics, such as long term durability have not been evaluated.

We only consider the performance of the voxels in a basic mechanical loading scenario and we do not consider the stability of these structures under shearing or lateral loads. To fully validate the voxel system, not only would more substantial mechanical testing be required, but a slate of other properties would also need to be tested. In addition, we would need to evaluate the voxels for long-term use. For example, mechanical fatigue, thermal stresses from different material types, and humidity resistance may all negatively impact the performance of the voxel approaches over many years.

Significantly, the fire resistance of most of the considered voxel approaches, especially the FRP-based ones, is too low for use as structural elements in a building, and future research would need to address this. The fire performance is a major constraint especially for polymer-based voxel systems, and may prevent future use of these systems. For this reason, this study also evaluates steel and wood-based voxel systems whose intrinsic fire performance may be more compatible with conventional building codes, although it is important to note that the structure of these may make them more susceptible to fire damage. While intumescent coatings or mineral-filled composite formulations may improve behavior, such strategies would increase cost and embodied carbon and may not achieve required structural fire ratings. Future work should explicitly assess code-compliant configurations within the present LCA framework.

Our path-planning tools currently do not account for the physics of the system, which may be a significant limitation for structures such as these which have large cantilevers during the construction process. While some prior work has demonstrated integrated CAD and FEA for voxel systems [162], these do not incorporate robotic path-planning and are likely not suitable for simulating structures at the architectural scale, where total voxel counts are extremely high. Potential integration of other work [80] [81] that has explored the generation of loading-aware robotic paths, or even hardware such as [120], which can measure loads in the structure, may be necessary for full scale deployment.

The robotic hardware system is also not at the requisite maturity yet for full-scale deployment, though there is active work towards this. The largest scale demonstrations of voxel assembly, from the MILAbot system, and from [71], both have limitations in the maneuvers that prevent efficient assembly of large structures, which is why, for this work, we simulate the performance using a vertically moving voxel feeder, which has not yet been demonstrated in hardware. Similarly, with the exception of the FFF-printed voxel, which is not a practical option, and the node-octet voxels, which have not been robotically assembled, all of the other voxels require some amount of assembly to go from 2D/2.5D faces into the voxel. In prior work, this has always been manual, but for real deployment, automating this would be necessary. Additionally, the delivery of voxels to the voxel feed or assembly robot is underexplored.

Based on the results from the assembly speed estimates, future work should also address more "complete" voxel modules. These voxels could potentially incorporate sheathing, insulation, or electrical or plumbing routing for a more direct-write assembly process. If this direction is further pursued, questions of how to properly seal a highly discretized structure become more relevant. Additionally, explorations of what the most appropriate size of voxel and size of robot may be relevant.

5.7 Conclusion

Depending on material and geometry selection, voxel-based systems were able to reduce the embodied carbon of the evaluated single-story building compared to concrete approaches. The plywood octet system demonstrated the lowest embodied carbon of the voxel approaches, with 17% the embodied carbon of the 3DCP approach, and 24% that of the precast concrete approach. The steel octet system had 36% the embodied carbon of 3DCP, and 52% that of precast, while slightly reducing the carbon footprint relative to the steel framing approach, at 91% the embodied carbon. On-site assembly time is competitive with existing approaches, at an average time of 99 hours for the two voxel approaches, and an average of 155 hours for the existing approaches, and the voxel systems are only outperformed by the large-panel based modular building systems. Total construction cost for the steel and plywood octet voxels is comparable to that of the concrete-based approaches, as well as the steel framing approach, although the stick-framing approach is substantially cheaper, at 70% and 63% the average cost of these other approaches, respectively.

This study shows that the robotic assembly of voxels has the potential to improve the sustainability and efficiency of construction compared to some existing construction approaches. For the voxel approaches, cost competitiveness depends on material selection and fabrication strategy. The voxel approach combines the mass-efficient benefits of framing systems with the automation benefits of 3DCP and modular prefabricated methods. Compared to the 3DCP approach, the voxel approach has lower embodied carbon, lower labor costs, and lower amortized equipment cost per structure, indicating the potential for discrete assembly in large-scale construction. Compared to modular prefabricated methods, such as prefabricated concrete systems or modular wood based systems, the voxel-based assembly approach is slower, but does not require large on-site infrastructure for installation and can support added geometric complexity without cost increases.

The results suggest that further exploring either a steel or wood voxel system would enable the most sustainability and cost benefits, and future work would focus on validating these projections with physical demonstrations. Future work could also explore where in the construction process adopting a discretized distributed robotic assembly system makes the most sense. The results demonstrate modular systems that leverage larger panels can achieve reasonable sustainability and cost metrics likely at an assembly speed that smaller voxel modules will not match. However, the ability to use smaller and simpler robots may have alternative benefits, especially toward more highly autonomous operation in remote or challenging environments, or towards creating reconfigurable or temporary structures.

As the work here explores a simplified single-story building, future work should also focus on projections for multi-story buildings or other large structures as a clearer path to scaling to large structures is one of the key motivators for exploring voxel-based systems. Scaling to multi-story structures will require evaluating metrics beyond mechanical performance, such as thermal expansion, fire safety, and relative humidity effects.

5.8 Voxel Cost and Emissions Estimation Methods

This section explains the methods and sources used to derive the cost and emissions estimates for the voxel approaches.

GFRP node connected octet (Figure 5-1A) achieves a compressive modulus of 2.6 MPa at a density of 5.8 kg/m^3, while the CFRP node connected octet reaches a modulus of 7.97 MPa at the same density. The carbon intensity of these material systems is determined using the carbon intensity figures for PEI from [29], glass fiber from [58], or carbon fiber from [100], leading to an embodied carbon estimate of 12 kgCO_2e/kg for the GFRP version and 19 kgCO_2e/kg for the CFRP version. [72] provides the cost for each of these at $0.01 per cubic centimeter at $0.03 per cubic centimeter, respectively.

The GFRP cuboct (5-1B) has a compressive modulus of 3.2 MPa and a density of 32.5 kg/m^3 [91]. The CFRP cuboct (5-1C) has compressive modulus 1.1 MPa and density 10.1 kg/m^3 [71]. We use the embodied carbon estimate for nylon 66 from [130] and the prior GF and CF estimates, leading to a value of 5 kgCO_2e/kg for the GFRP lattice and 13 kgCO_2e/kg for the CFRP lattice. The cost of the GFRP lattice is $1.38 per face, excluding mold cost, while the CFRP lattice is $7.43 per face [71].

The bio-cuboct (5-1D) has a modulus of 9.39 MPa and a density of 73 kg/m^3. Based on the values reported in [67], the voxels have an embodied carbon equivalent of approximately 5 kgCO_2e/kg. Though this value is slightly higher than our estimate for the conventional GFRP voxel, the bio-cuboct has superior specific modulus performance. Because the material formulation for the bio-cuboct is discontinued, and because [67] does not provide a cost estimate, we re-use the cost per face from the [91] GFRP cuboct, which is likely a low estimate.

The reinforced Kelvin lattice (5-1F) has a modulus of 34 MPa and a density of 151 kg/m^3. It is worth noting that this lattice has a relatively smaller lattice pitch (17.2 mm) compared to the other considered approaches, and is designed for incrementally adding new faces of material as opposed to entire volumetric elements. However, for this work, we treat it as though it is also a voxel-based lattice decomposition. We estimate the embodied carbon of this approach by first estimating the embodied carbon of the mild steel sheet stock used to make the voxel parts. We start from the reported CO_2e emissions for producing hot-rolled merchant bar quality stock from Gerdau [153], which includes GHG emissions from initial steel production, and assume a cold-rolling finishing step consuming energy at rates described in [147] with an emissions factor based on the current Massachusetts grid embodied carbon per megawatt hour value of 418.666 kgCO_2e/MWh [7]. We then factor in the CO_2e emissions cost from cutting the material on a 1.2kW laser cutter (xTool MetalFab), as well as the wasted material from the lattice cut pattern— for this shape, we estimated a maximum packing density of 23%. From this, we arrive at an embodied carbon estimate of 4 kgCO_2e/kg of produced voxel. We estimate the cost of the voxels to be $7 per kg of produced voxel based on the cost of sheet stock from [15] with the cost of electricity to cut the material and one technician paid at $25 per hour operating the machine for its entire run time.

The aluminum cuboct (Figure 5-1G) is made directly from an aluminum substrate PCB (approximately 1.5 mm thick aluminum core with insulation, copper, and solder mask to a total 1.6 mm thickness) and reaches a modulus of 24 MPa at a density of 30.75 kg/m^3. We estimate the embodied carbon of this approach by combining the base embodied carbon associated with producing the aluminum at 11.8 kgCO_2e/kg, based on figures from [173], with the cost of producing a single layer PCB, which we estimate at 18.37 kgCO_2e/kg based on the figures in [110], leading to a value of 30.17 kgCO_2e/kg of produced voxel, the most carbon-intensive of the considered voxel approaches. The cost of this system is $0.35 per face, based on quotes from JLCPCB for 1000-part order quantities.

The expanded Miura (Figure 5-1H) achieves a compressive modulus of 50 MPa at a density of 110 kg/m^3. We estimate the embodied carbon of this approach by combining the embodied carbon from producing the base materials with the additions from laminating the sandwich panel and processing it on a Zund flatbed cutter, determining the machine running cost with the same method as the laser cutter, assuming 67% packing efficiency. From this, we get 28 kgCO_2e/kg of produced voxel at a cost of $27.70 per kg.

The embodied carbon of the PLA octet (Figure 5-1E) approach is based on the value provided in [57], with an overhead added based on printing the voxel on a Prusa Mk4 with an operator in attendance for 0.5% of the print time based on our measured print times versus print loading and unloading times. This results in a total embodied carbon of 3 kgCO_2e/kg. The cost is taken from the Prusa slicer, at $0.62 per voxel. The steel octet voxel embodied carbon is calculated in the same way as the Kelvin lattice, but with a maximum packing efficiency of 45%, resulting in 2.4 kgCO_2e/kg of produced voxel. We estimate the cost at $8 per kg of produced material. The plywood octet voxel is calculated similarly, starting with the emissions factor from [173], and processed with a 120 W CO_2e Epilog laser cutter, resulting in 1 kgCO_2e/kg of produced voxel, and a cost of $8 per kg of produced material.

5.9 FEA Simulation and Experimental Validation

We recreated physical and virtual three-point bending tests to verify the analogy claimed in the paper: that the physical cellular materials we propose can be simulated as a digital continuum material. We selected the three-point bending test as it also challenges the material assumption in a more realistic way than a uniaxial compression test. A three point bending test is not fully tension dominated, but rather is a good combination of both compression and tension modes of work.

To this end, we selected three materials and four different beam architectures. Using the 3D printed edge-connected octet lattice, we produce two beams, one in PLA and another in a PLA/PHA blend. The compressive modulus of each is measured using a single unit cell specimen, and extrapolated as the bulk material property for simulation. We produced an additional beam using the node-connected octet lattice, but using a non-fiber reinforced Nylon. A 2x2x3 block sample was compressed to determine its compressive modulus and

Table 5.2: Comparison of simulated and experimental 3-point bending tests

	PLA Octet	**PHA Octet**	**Expanded Miura**	**Nylon Node Octet**
FEA stiffness (N/mm)	137	184	281	26
Measured stiffness (N/mm)	128	152	221	44
Compressive modulus (MPa)	15	25	45	1.5
Density (kg/m^3)	81	82	110	4.5
Lattice pitch (mm)	65	65	66	76.2
Beam dimensions (L×W×H)	6×1×2.5	10×2×6.5	8×1.5×2	8×2×3

weighed to determine its weight. From the literature, [139] reports three-point bending test results for the geometry we evaluate in this paper, and these are taken as the experimental results for that beam. The dimensions of each beam we tested are listed in Table 5.2.

We chose these geometries as they span a sweep of the voxel types evaluated in this paper. The PLA and PHA octet lattices leverage the compounded assembly, as previously described in this work, where pre-connected blocks (e.g. 1x3 strip of voxels) are assembled into larger structures primarily connected vertically through snap fit joints, with dove tail features providing additional lateral/tensile stability. The interleaving of the compounded blocks provides good structural stability and capacity to support bending loads. The node octet system then is connected voxel-by-voxel, and so is less likely to have joints with significantly different strengths and stiffnesses throughout. These voxels are connected via 0-80 screws and nuts. The expanded Miura voxel is a plate lattice system, and the beam assembly incorporates continuous sheets that are loaded under tension during bending, providing a comparison point to a specimen with complete continuity. Voxels are connected via blind rivets.

In Abaqus, we model the three-point bending test by representing the specimen as a rectangular continuum with the corresponding elastic modulus E and Poisson's ratio ν , as specified in Table 5.2. The cylindrical supports and the punch are modeled as discretely rigid bodies, assuming no deformation. The interactions are defined with tangential penalty friction using a coefficient of 0.15, and normal interaction is set as hard contact. A prescribed deflection is applied to the punch, and the resulting reaction forces are recorded to replicate the measurements obtained from the Instron load cell in the physical experiments. Displacement and reaction force data are extracted and compared with the experimental results. For all Abaqus simulations, units are defined as mm, N, and MPa.

We see relatively good agreement between the simulated beam performance and the measured beam performance, in terms of the reaction force to displacement ratio. These are plotted in Figure 5-22. For the expanded Miura beam, the measured result is 221 N/mm while the simulated result is 281 N/mm. For the PLA octet beam, the measured result is 128 N/mm and the simulated result is 137 N/mm. For the PHA octet beam, the physical result is 221 N/mm and the simulated result 281 N/mm. For the nylon node octet, the physical result is 44 N/mm and the simulated result is 26 N/mm. This significant under performance of the FEA model is likely because of the softness of the bulk material, as the

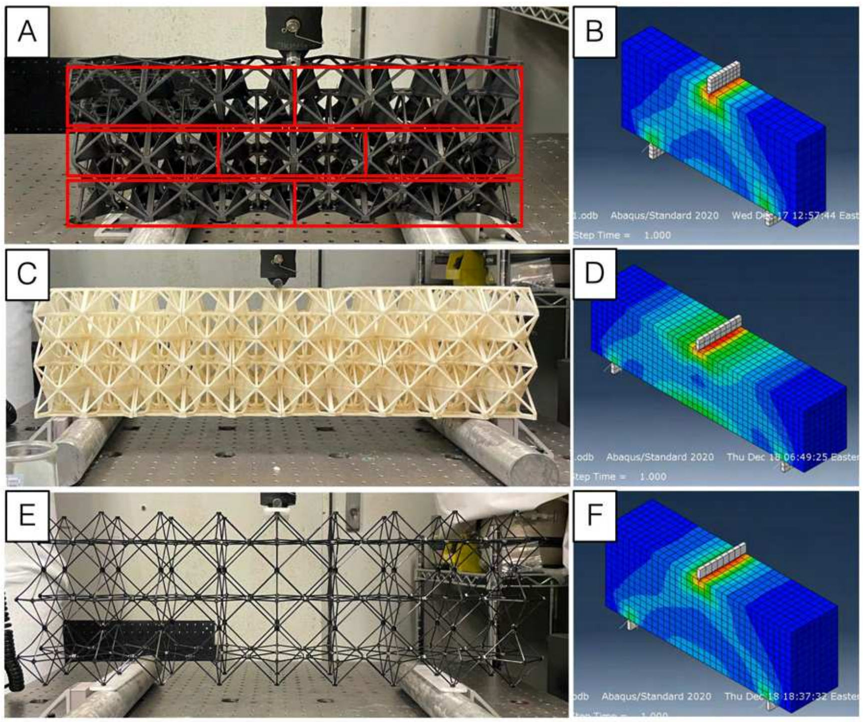

Figure 5-21: Three-point bending testing physical set up and simulated set up. A) The PLA octet beam, with separate voxel blocks highlighted in red. C) The PHA/PLA blend octet voxels. Like the PLA octet beam, this one also consists of stacked and staggered compounded voxel elements. E) The nylon node octet beam. B), D), F) are the simulated versions of the beam. A reference for the expanded Miura beam is available in [139].

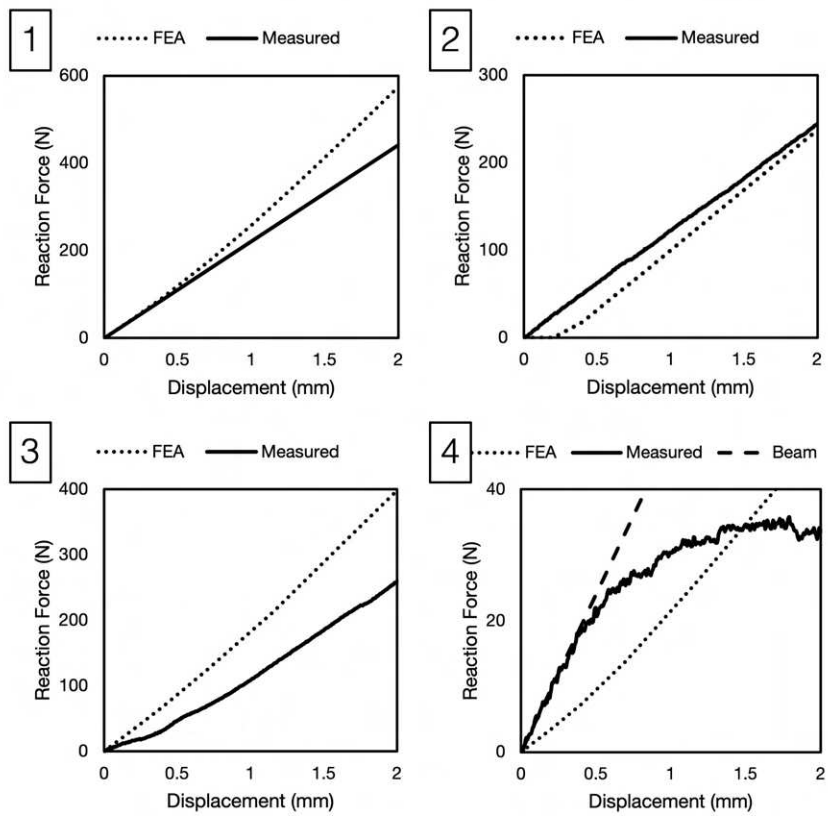

Figure 5-22: FEA and physical force-displacement plots for the four tested beams. 1) Expanded Miura beam. 2) PLA octet beam. 3) PLA/PHA octet beam. 4) Nylon node-octet beam. Note that for the Expanded Miura, the measured data is a plot of the slope of the raw data available from [139]

.

modulus is quite low. If we analytically calculate the expected bending stiffness EI based on our measured modulus, as well as the bending stiffness from our 3-point bending test, we see much closer results. Using this analytic bending stiffness EI, we predict a force-displacement ratio of 48 N/mm (this is plotted in Figure 5-22 as the dashed "beam" line). For all of these results, the physical behavior of the discretely assembled beam is relatively well aligned to that of the simulated continuum beam model. In addition to the existing literature on discrete lattice vs. continuum lattice performance, these results suggest that our simplified simulation model reasonably captures the real physical performance of voxel assemblies in more complex loading scenarios.

For both the structural archway and the three-point bending tests, the simulation code is available here: https://gitlab.cba.mit.edu/alfonso/abaqus-python-code-for-comparative-evaluation-of-da-lattices-for-construction/-/tree/main.

5.10 Key Parameters of Building Approaches

The key parameters of the considered building approaches are summarized in Table 5.3 on the following page.

Table 5.3: Comparison of building approaches with material, cost, and performance metrics.

Approach	Density (kg/m^3)	E (MPa)	Pitch (mm)	Voxels (#)	Wt. (kg)	EC ($kgCO_2$)	Mat. Cost (USD)	Fab. Cost (USD)	Total Cost (USD)	Time (h)
Alum. Cuboct	30.8	24	150	10,416	1303	40,849	21,580	4,991	26571	95.8
GFRP Node-oct	5.8	2.6	76.2	267,720	1102	14,502	46,633	7,917	54,551	144.4
CFRP Node-oct	5.8	8.0	76.2	226,776	627	13,622	111,090	7,208	118,298	132.6
GFRP Cuboct	32.5	3.2	75	464,016	7150	34,688	2,758,734	9,320	2,768,054	167.8
CFRP Cuboct	10.1	1.1	300	12,816	4097	56,367	325,129	11,458	336,587	203.3
Steel Kelvin	151	34	17.2	5,254,368	6044	28,259	841,48	4,101	88,249	81.0
Bio-Cuboct	73	9.4	75	80,016	6552	34,306	1,381,008	6,575	1,387,583	122.1
Exp. Miura	110	50	66	36,384	2402	23,024	70,590	4,299	74,888	84.3
PLA Octet	82	25	65	44,328	3157	11,904	72,118	4,741	76,859	91.6
Steel Octet	116	126	100	15,624	1921	6158	16,776	4,252	21,028	83.5
Plywood Octet	15	8.1	100	69,624	1246	2874	13,301	6,176	19,477	115.5
Steel Frame	–	–	–	–	2074	6609	15,859	10,999	26,858	282.1
Stick Frame	–	–	–	–	1150	1933	4,983	4,981	9,964	108.5
3D Concrete	–	–	–	–	46,267	17,338	12,973	7,616	20,589	119.7
Precast Conc.	–	–	–	–	42,114	11,899	23,681	2,677	26,358	54.3
CMU	–	–	–	–	48,022	14,005	14,250	9,823	24,073	209.0

Chapter 6

Multi-Material Multi-Functional Voxel Assembly

6.1 Introduction

Every element in a built structure implicitly encodes logic about its structural performance, assembly sequence, manufacturing process, and functional behavior. The overall dimensions of a timber beam, for example, are informed by the loads it must carry, how it fits into typical fastening and framing practice, supply chains, on-site logistics, basic manufacturing process constraints, and how well it supports a given design vision. A stick framed house is not built from pieces of wood with arbitrary cross sections and lengths, but rather a limited set of standardized parts that reflects the structural, functional, economic, and assembly-based constraints and needs of the project. This information is all implicitly encoded into the geometry of the building element, and it is decoded primarily by humans who understand how to design, verify, and build with these structural elements. Because this information is standardized, it enables the mass production of the parts and their use in a variety of different structures in different environments with different functional requirements.

Because these conventions have been developed around human building practices, it is difficult to port them to machines or robots, which lack the dexterity, sensing, and adaptability we have. This can be managed by either making larger, more complex, and more expensive robotic systems, as is being done in the various industrial arm assembly projects [113] or the vast amount of humanoid robotics research, or we can take the opposite approach and build materials specifically for simple robots to work with (as we have been seeing in this thesis). The key observation here is that the implicit design intelligence in building materials for humans still needs to exist somewhere in the system in a way that is accessible to the robots. This can be done by considering these various dimensions of a structural element and explicitly embedding them in the design of the building material. We can develop building materials that describe their own structural performance and determine their own assembly.

So: we have progressed through many generations of voxels that all essentially circle around this goal of a generalist large-scale building system, with each generation of voxel adding or emphasizing different critical (and not so critical features). The goal of this chapter is to reach closer to the synthesis of these into a coherent and unified building system: robotically assembled, structurally performant, mass-manufacturable voxels that can be made from any standard engineering material into useable structures at practical assembly speeds and costs, with low embodied carbon footprint.(Of course, the underlying goal of a parametric and autonomous building system for almost anything is not achieved, but I hope that this chapter provides an outline for how this might be achieved).

The core contributions of this section are:

- A multifunctional building block system for large-scale structures that can be assembled from a flat-packed feedstock that can be manufactured from a wide range of material types
- A distributed, mobile robotic assembly system developed to assemble these blocks, both from 2D faces into building blocks, and from building blocks into larger structures.
- The evaluation of these systems through a set of diverse demonstration structures: the construction (and reconfiguration) of a prototype house, a set of miniature electric cars, and furniture.

A physics-aware planning system and robot coordination tool has also been developed, but this is primarily work of my collaborator Alessio Zazo, and so the details of this will not be a focus of this chapter, though I do summarize it in the following section to give a more complete overview of the building system.

6.2 Methods

6.2.1 Planar Voxel Blocks

We have developed a design pattern for the scalable production of high-performance voxel building blocks that is compatible with a range of material types and manufacturing processes. It starts from a flat feedstock, which enables efficient transport pre- and post-assembly as well as compatibility with a wider range of sheet materials and processing methods. This 2D voxel decomposition is designed around low-degree of freedom assembly: the voxel block is assembled almost entirely from the vertical direction, fully constraining the assembly using a final part with an orthogonal installation direction. This low-DoF assembly is favorable for both automated assembly (robotic or otherwise) and for increasing the throughput of manual assembly. We demonstrate this voxel system using metals, plastics, and wood, and show that we can achieve a wide range of mechanical properties.

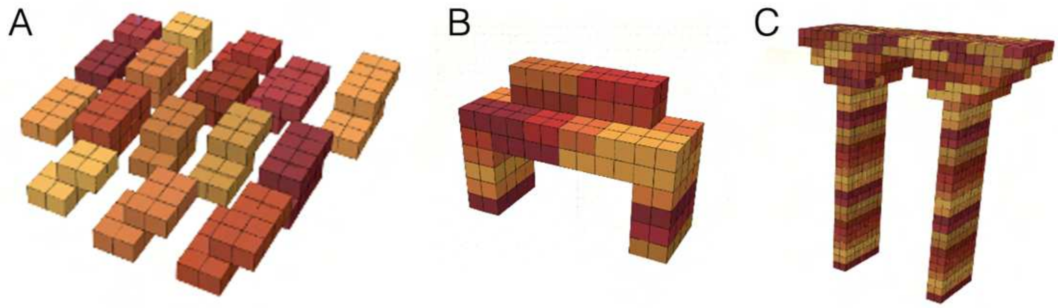

Figure 6-1: Simplified block views of voxel structures. A) The library of compounded voxel blocks used in the assembly software. B) Example bench, re-computed with this library in a structurally stable configuration. C) Example archway with different block color highlighted. Images from Alessio Zazo.

Implementation Details

Grouping individual lattice unit cells into larger block configurations, such as a 2x2x1 arrangement, or stacked and staggered arrangement, as shown in Figure 6-1 simplifies and speeds up the robotic assembly process. So that these blocks are fast and easy for the robot to assemble, they are designed for vertical assembly. Blocks are added to the structure by placing them on top of other blocks, with the pyramidal faces acting as structurally active alignment features into the voxel block below (see Figure 6-10). In this iteration, a snap fit at the base of the pyramid finally constrains the position of the block, while the pyramidal features provide shear stability, and dovetail features along the upper face transmit lateral loads. By tiling these blocks in space such that they overlap with each other, we offset the seams of these joints for better load transfer.

Chapter 4 introduces this staggered block concept and explores it through an FDM 3D printed block system, which limited the robustness of the structures and the potential for mass-manufacturing. Chapter 5 considers a similar layout but utilizes separate truss elements to constitute each voxel. While this supports fabrication from planar feedstock, it would be difficult to automate the 2D-to-voxel block process, as it requires many dexterous operations. The robotic voxel assembly system in [71] similarly does not support automated assembly from 2D to voxel units (and this would be non-trivial to implement).

An automated method to go from 2D elements to volumetric elements provides two critical features for the practical deployment of a block-based assembly system. The first is the ability to flat-pack constituent parts to the target build area. This is highly relevant for in-space assembly tasks, a prime application point for distributed assemblers [137], as launch payloads for large assemblies are primarily volume limited. This is also relevant for on-site construction automation, as it is similarly preferable to transport a flat-packed set of parts than the equivalent final built volume. The second key feature is that an automated assembly system can enable substantially improved production throughput. While many prior block-assembly works report high assembly speeds, they typically draw the assembly boundary around the block-to-structure process such as in [71] or [89]. Though this can be valid in many deployment scenarios, such as if reconfiguration from an existing library of blocks is the dominant mode of assembly, it can also obscure a likely bottleneck for many assembly scenarios, such as for on-site construction or in a manufacturing setting, where we would most likely deploy the full face-to-block-to-structure pipeline.

To do this, we decompose the grouped voxel block into flat faces that can be assembled using low degree of freedom operations: all parts are installed from the same direction, with a final part that is inserted orthogonally to fully constrain the assembly. This is done using a flexural latching feature. A long flexure is cut into the wall panel parts, such that we can expect reasonably easy deflection in a wide range of materials. When the top face of the block is placed, this flexure deflects inwards. As the top is slid backwards, the flexure will eventually latch into place, preventing the block from self-disassembling. A detail view of this feature is shown in Figure 6-7.

We can further simplify the assembly by reducing part count. This is done by considering the intersecting faces of an edge connected octet lattice, simplifying into solid planar parts, and removing select planes according to most likely loading scenarios. In this way, we can build out our layers of voxel groupings from three part types that are repeated as necessary to achieve the appropriate dimensions: a width part, a length part, and a top part, as shown in Figures 6-2 and 6-5. In a 3x2x1 block, for example, we would need two length parts (three voxels long), three width parts (two voxels wide), and a single top part with a 3x2 pattern to constrain it. Staggered blocks are then made by pre-stacking constituent blocks at the appropriate location.

This decomposition also nests effectively on a 2D plane, which means that we can process it using cutting methods without wasting substantial material. The nesting efficiency will of course depend on the lattice pitch relative to the sheet stock size as this could substantially impact cutting margins, but assuming a 10 mm margin along the sides of the part, we hit a continuum packing efficiency of over 96%. The top-type part has voids and, if fabricated as a single part, is designed to be made via a net shape process, such as 3D printing or injection molding. By default it is a 2.5D design, but the non-planar elements function as spacers and are not critical. If necessary for the application, the top part may be directly fabricated as one part from sheet stock, which will produce substantial off-cuts, or it may be fabricated in sections or with stiffeners, which will introduce pre-assembly steps, but can minimize material usage, an example of which is shown in Figure 6-4.

For a truss lattice, we can directly hollow out the faces of width- and length-type parts and use a net-shape fabrication method. As with the top-type part, this is also portable to subtractive processes without substantially increasing fabrication waste if we are willing to introduce additional assembly steps to the process. This gives us three main levers with which to modify the performance of the block itself: the base material, the thickness of the parts, and the overall fill volume. This represents a substantial design space, which is discussed in more detail later in this chapter. In this thesis, we show versions of the lattice

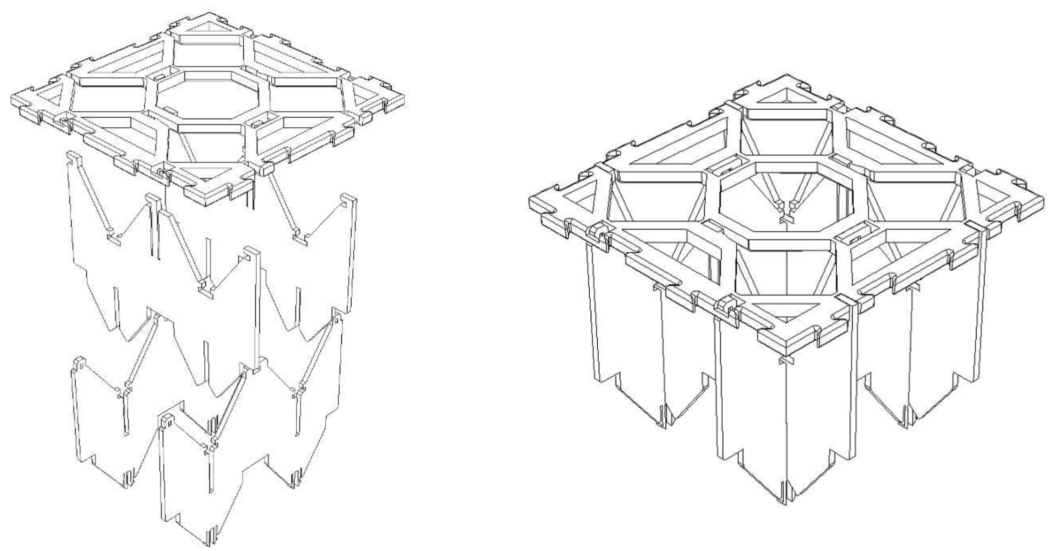

Figure 6-2: (Left) Assembly sequence of a 2x2x1 voxel. (Right) The fully assembled voxel.

made from plywood, aluminum, and various plastics, with different material thickness and fill volumes.

To form base layers, we add two families of base voxel parts. The first is shown in Figure 6-3 C and D and is essentially a version of the voxel block without the the pyramidal or snap features. This way, it can rest flat on the ground or provide a base with protruding sharp objects. A second version of a base voxel is shown in Figure 6-8, which incorporates the dove tail joints for both the top and the bottom surface. This allows tensile load transmission along the bottom layer of a voxel structure, which is very significant for bending performance. This voxel is sized at half the height of the lattice pitch, so as the occupy less space.

(I will pause here to note that while in this thesis, parts with voids are primarily fabricated through additive processes, this does not mean in real deployment these parts would be limited to only plastic or FRP parts! There are a wide range of mature (and immature) commercial processes which could produce these parts in a range of materials that are simply beyond the scope of this thesis. For example, custom-profile aluminum or steel extrusion processes could produce any of the parts used here, but because of the initial tooling cost, this would be more appropriate for a commercial and not an academic setting. Alternatively, the continuous carbon fiber process employed in [46] could be applied to these parts, but would require a dedicated machine for fabrication and so is once more out of scope.)

I will also show here a few of the design iterations that led to the current planar voxel block, as this may be useful to the key audience of this thesis, the future CBA student.

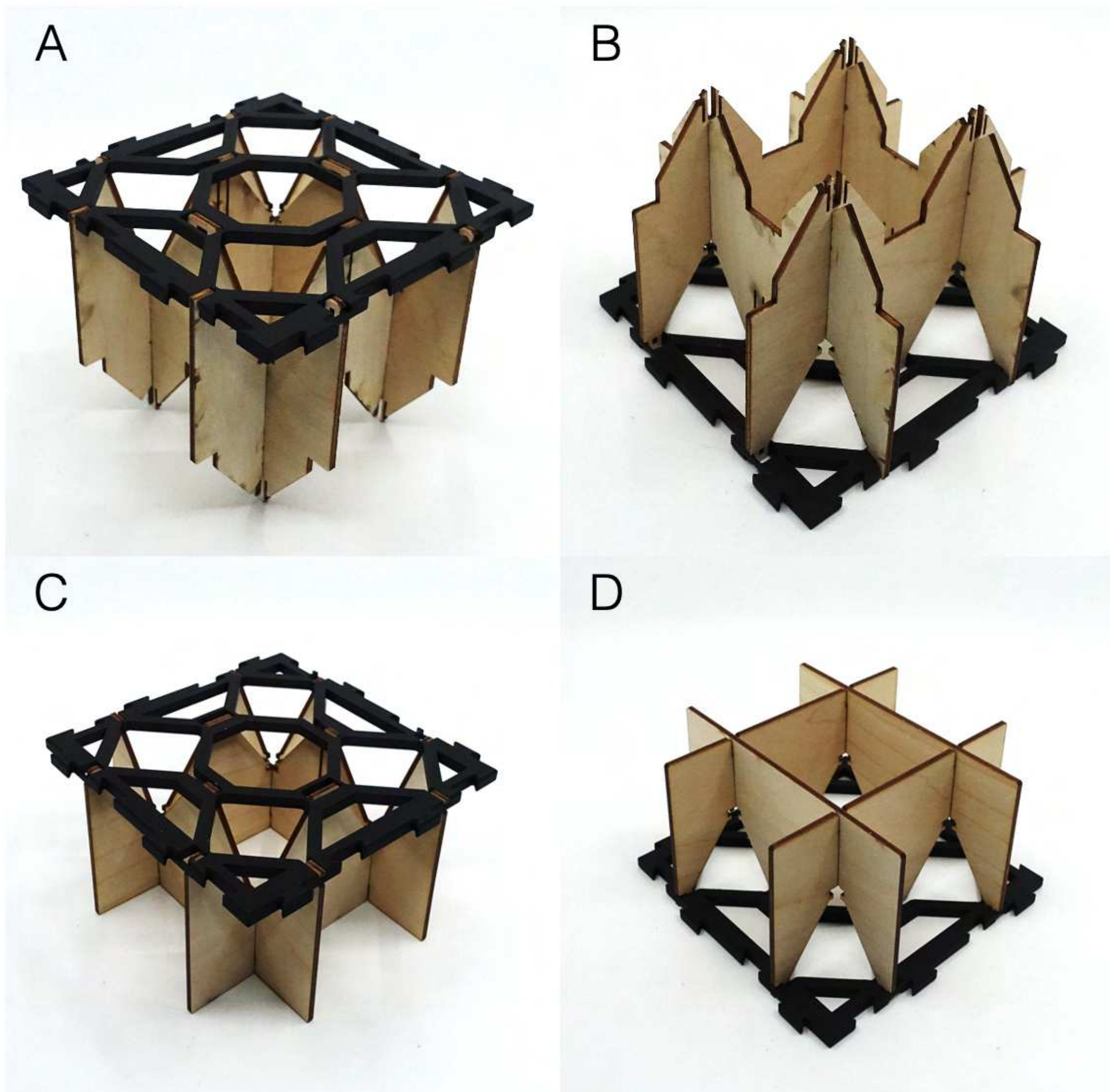

Figure 6-3: A) Upright 2x2x1 wood voxel. B) Upside down voxel, showing the snap fits. C) and D) Views of the flat-base style voxel, used for the first layer of a structure (or any other layer where a flat surface is desirable).

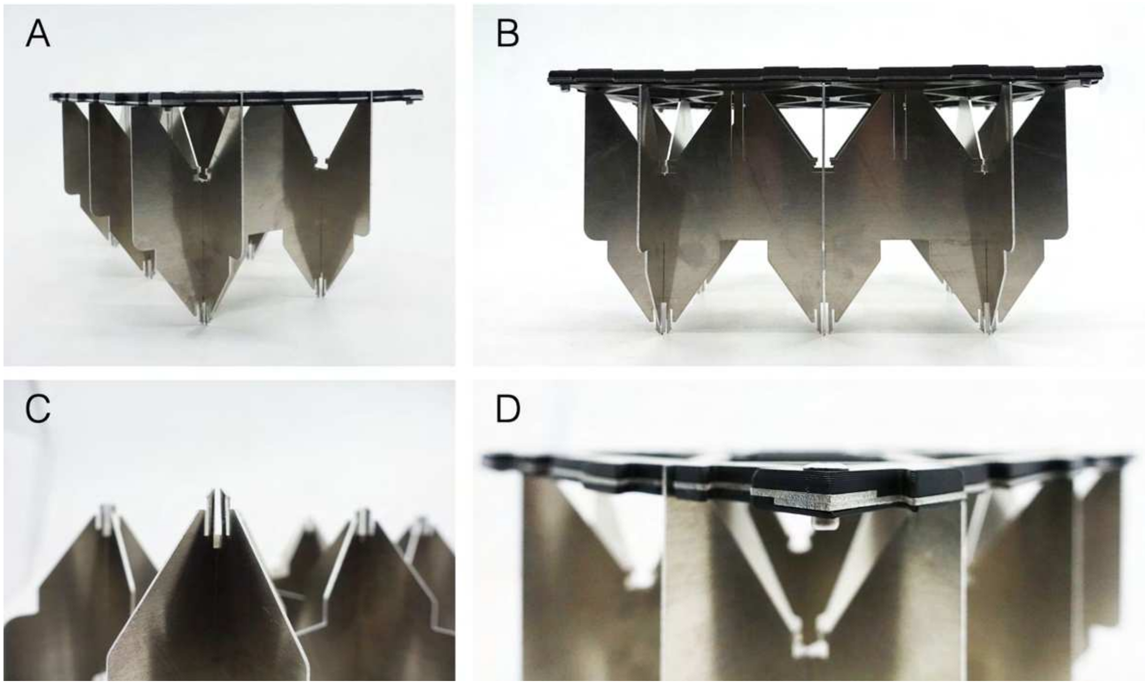

Figure 6-4: Views of a 2x3x1 aluminum voxel. A) and B) side views. C) Detail view of the snap-feature. D) Side-view of a reinforced top part.

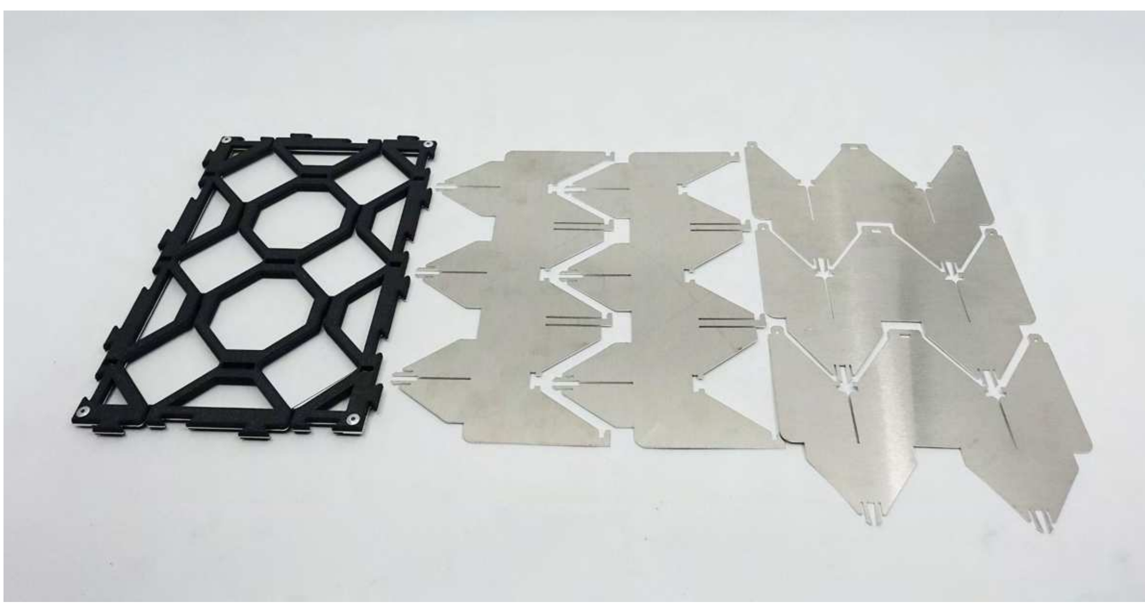

Figure 6-5: Flat layout of the part of a 2x3x1 aluminum voxel block.

Figure 6-6: A 3D printed PLA compounded voxel block with hollow, or truss, walls.

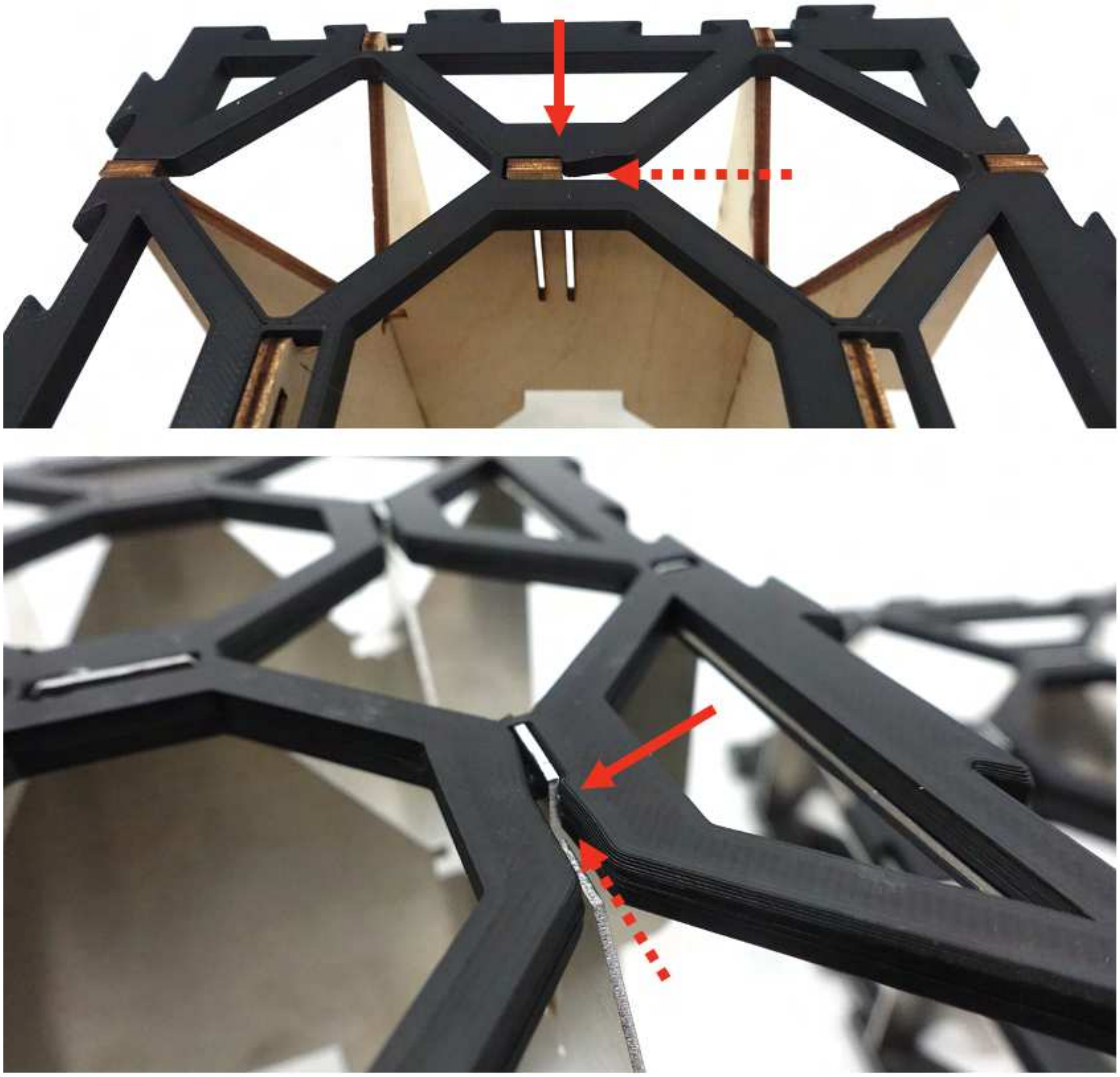

Figure 6-7: Close-up view of the flexural installation point of the voxel top surface. Solid line indicates installation direction while the dotted line points out the lateral egde that holds the voxel parts in place.

Figure 6-8: A second voxel-base style with the dovetail feature along the bottom as well; this design is better suited for bending loads as the dovetail layer provides a surface to transmit the tensile load over.

Figure 6-9: A half-base style plywood voxel viewed from below. The dovetail joints transmit tensile loads along this surface.

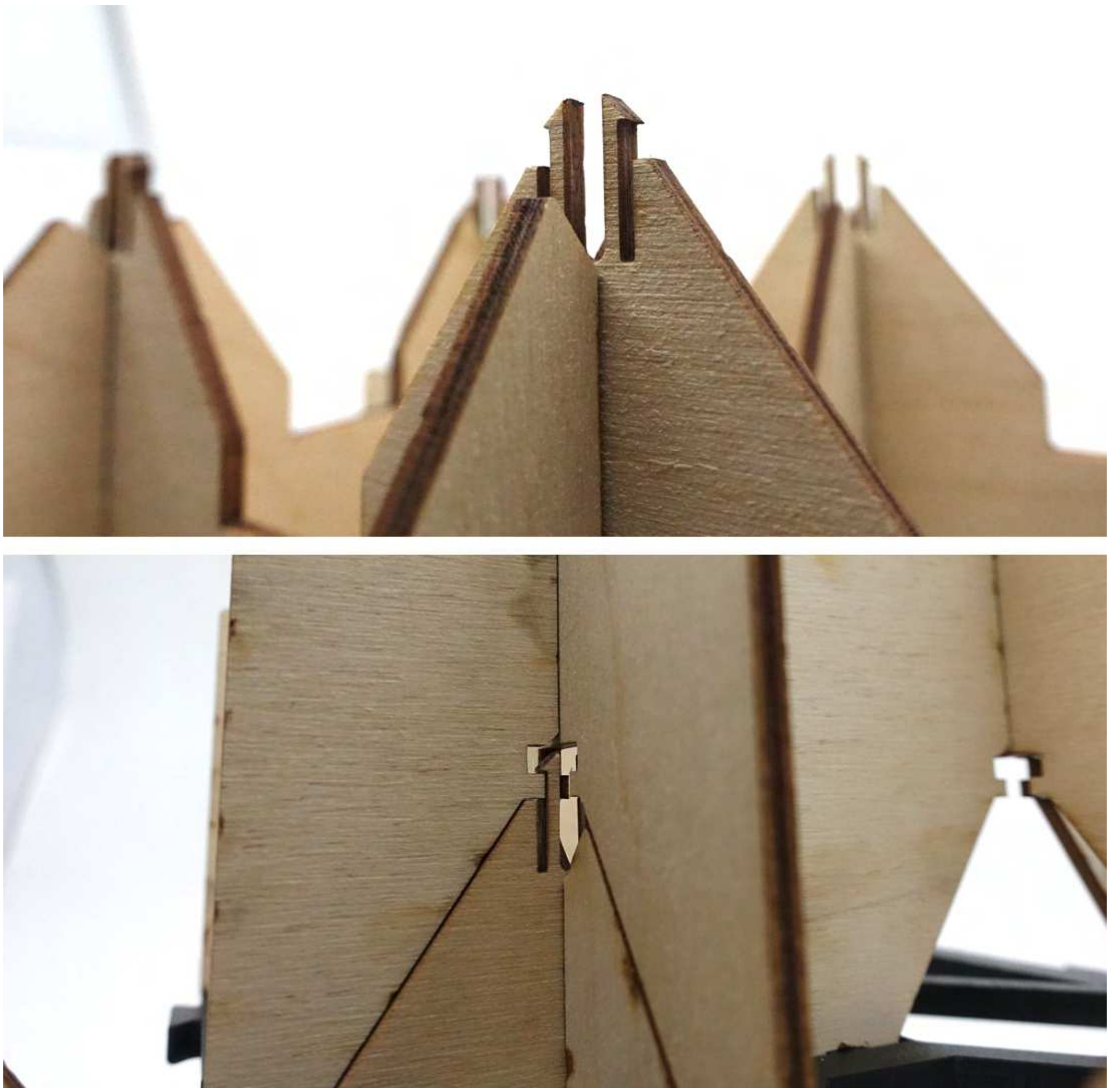

Figure 6-10: Close-up view of the snap fits on a wooden voxel.

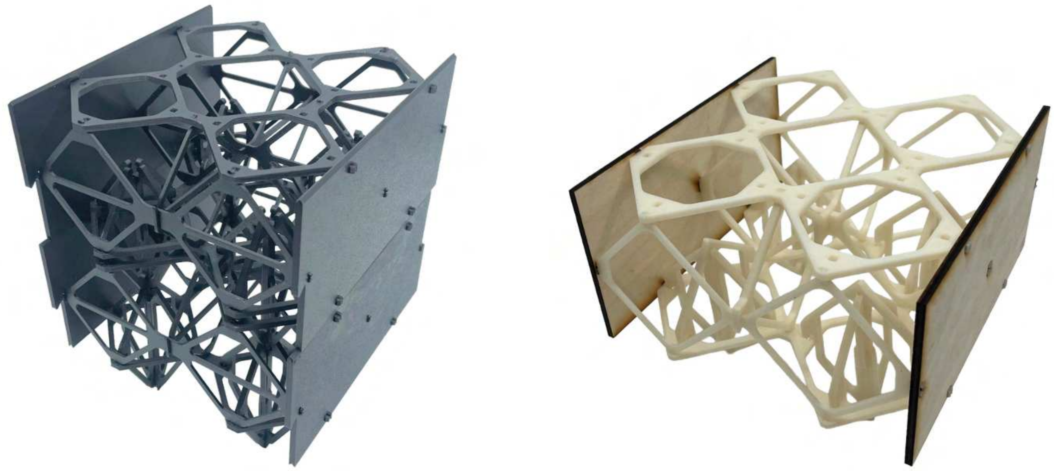

Figure 6-11: Some earlier attempts at the robotically assemble-able flat stock voxel which employed external walls as the final locking mechanism. The 3D version of the wall is lipped to provide a decent seal.

Figure 6-11 shows two generations of a vertically assembly voxel system, where all of the parts, except for the walls, are installed vertically, with the walls popped on laterally as a final installation constraint. There were a handful of issues with this generation of the design: the large quantity of parts, many of which are unique, and the reliance on snap fit joints between the faces introduces constraints on load-bearing, base-material options, and joint footprint. What this generation has that is quite nice is an integrated wall panel system compatible with robotic placement. I also realized that while structural loads were primarily being transmitted through the walls, the robot's weight was loading only onto the much more compliant lattice underneath. This prompted the above described design, where we much more heavily leverage a panels based system.

6.2.2 MILAbot v2

Many large-scale digital fabrication systems are ultimately constrained by the size of the machine itself. While mobile methods do not necessarily face this constraint, they often introduce higher costs and slower build speeds coupled with less reliable and less precise performance. To avoid this, we adopt a relative robotic assembly method (as discussed ad naseuam in this thesis), where we deploy mobile robots that locomote and work directly on the structure that they are assembling, deriving their global accuracy from their relative relationship to the underlying structure. This allows us to build structures much larger than the robot itself without introducing the need for advanced sensing and feedback systems, as

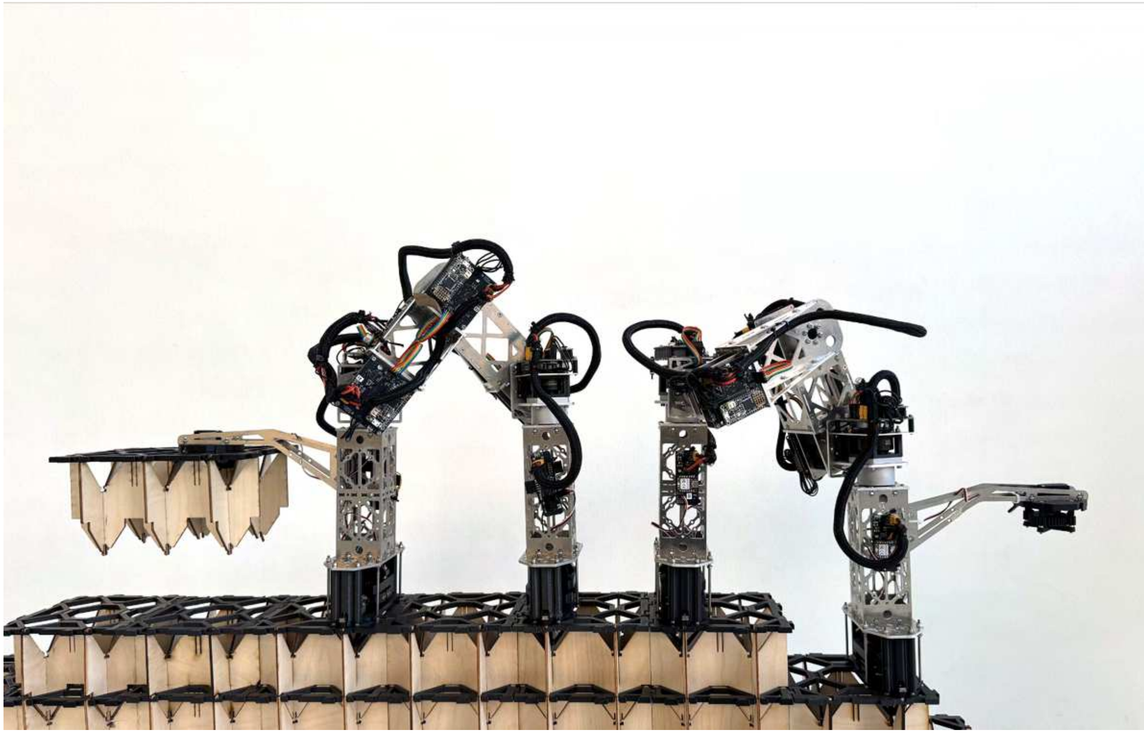

Figure 6-12: Two MILAbots on a wooden voxel structure

the accuracy of the robot derives from the predictable and inherently mechanically error-correcting structure underneath it.

For this work, we extend on the system introduced in [163], the Modular Inchworm Lattice Assembler robot, or MILAbot, to improve its reliability and scalability. The MILAbot v2 is built around a custom actuator module that uses a BLDC gimbal motor and an off-the-shelf harmonic reduction for closed loop control. The robot weighs 4.5 kg, and with a single payload gripper, carries a payload of up to 1.75 kg (depending on the moment arm extension). The maximum joint speed is 120 rpm, which corresponds to an average time of 9 seconds per movement type, where a movement is defined as e.g. a full step forward, a voxel-block placement, etc. This speed is relatively consistent as most of the movements traverse a similar amount of space.

MILAbot v2 Implementation Details

This subsection details the implementation of the MILAbot V2 and is intended to give a perhaps suitable overview to someone looking to recreate the system.

We use a GM3506 gimbal motor from iFlight coupled with a harmonic reducer, either the PGFUN NEMA17 100:1 harmonic reducer or the SITO drive HD-08 100:1 harmonic reducer. The NEMA17 reducer is a bit cheaper and much heavier, though which one we

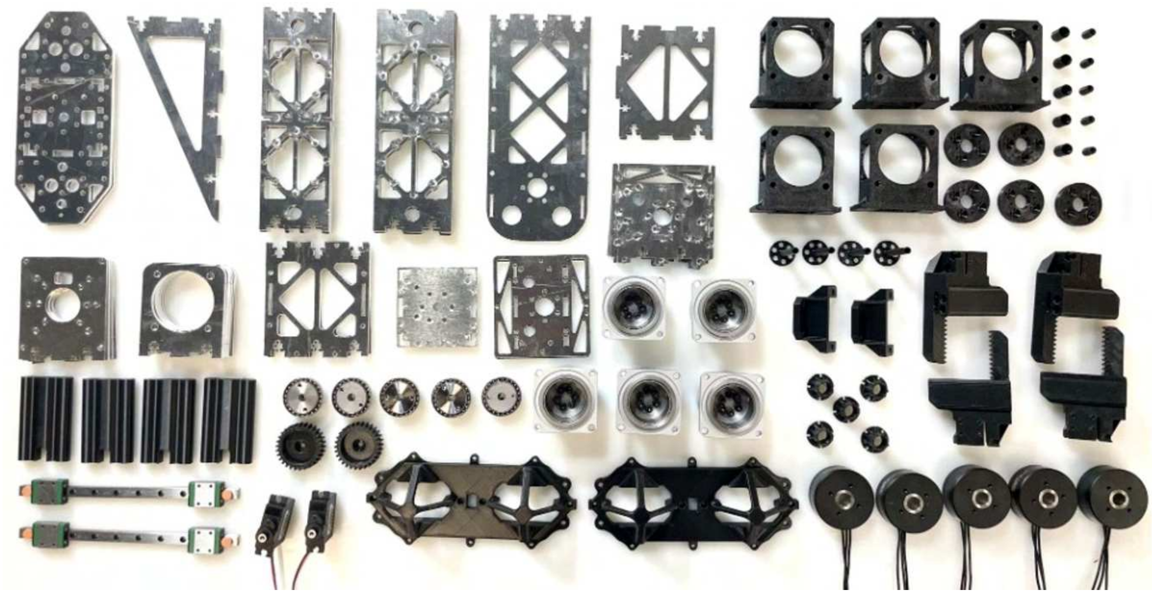

Figure 6-13: All of the parts (aside from PCBs) for a MILAbot build.

use is primarily driven by lead times and availability. In the MILAbot v1, we used 3D printed reductions to avoid this supply chain dependency, but within this packaging size, we were at the bleeding edge of performance for plastic 3D printed parts (intermittent 15 Nm loads)— this of course leaves a wonderful open spot for future work: the development of a non-plastic self-manufacturable reduction from 10:1 to 150:1, possibly backdrivable, and not too expensive. In the MILAbot v1, we implemented feedback on the motor shaft and on the output shaft by placing these off-axis from one another. This introduced collisions and motion constraints that in retrospect I was perhaps overly cavalier about. For the MILAbot v2, we move the feedback so that they are on the same axis, so as to satisfy our voxel-based packaging constraints.

This is done by attaching a 3D printed shaft to the output of the harmonic drive and running it backwards through the hollow shaft of the motor (we don't need continuous rotation for the robot anyways, so this is no loss). For closed-loop control of the motor, we use an AMT encoder 102V, which is sized such that we can attach it to the output of the motor and still maintain a hollow shaft space that the harmonic output shaft runs through. To this, we attach a diametrically magnetized 6 mm diameter magnet whose position is then read by the AS5047 magnetic encoder.

The motor is driven using a DRV8316 and SimpleFOC [160] running on an Adafruit Feather M4 Express, which is a SAMD51 based microcontroller development board. Development boards are in fact convenient to use, but, to the future CBA student looking for nice task to do, perhaps lay this all out as a single board with tighter packaging, something which I have so far not been able to bring myself to do, but which would improve the overall envelope of each module and likely reduce BOM cost per board by about 10-15 USD.

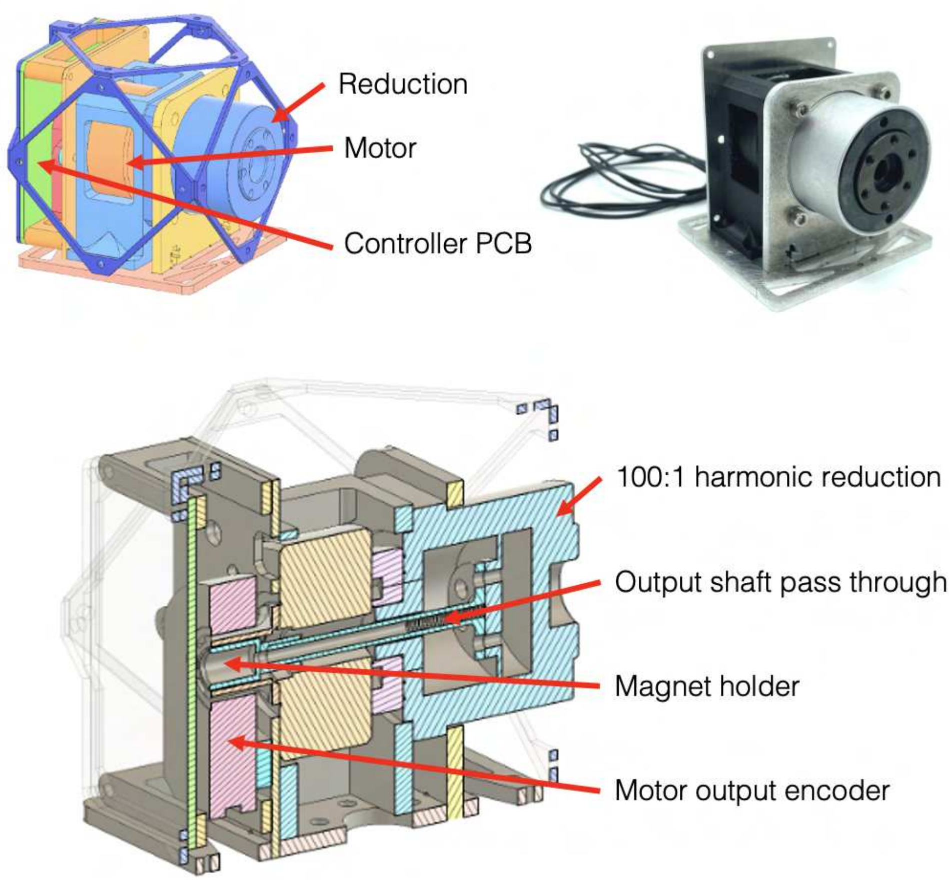

Figure 6-14: The motor module in the MILAbot v2. The lower image shows a cutaway view of the module model to show the feedback pass-through mechanism.

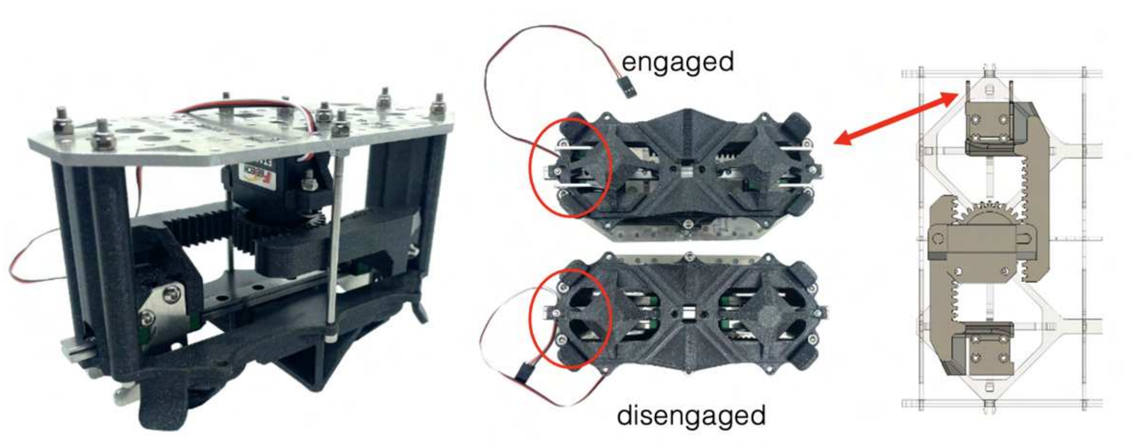

Figure 6-15: Locomotion gripper of the MILAbot v2 with the engaged/disengaged states noted.

We run a position-based PID loop on top of SimpleFOC to drive the actuator to desired output positions. The actuator module is set up to receive commands over I2C (desired position and such) and to report back encoder position readings. This represents a minimal, though functional, communications set. I don't expose e.g. actuator speed, PID gains, etc. for online tuning because it seems unnecessary. It is left as an exercise for the reader, if it is truly desired.

The MILAbot also utilizes spacer modules. For the v2, we have three types: a single voxel height spacer, a double voxel height spacer, and spacers to the gripper feet. Why multiple modules when one would work? Because it's faster to assemble. If you are working on a self-assembling version of this, then switch to one passive-style spacer, easy. These spacer modules are cut from a mix of 1.5 mm and 3 mm 6061 aluminum. In the MILAbot v1, I did some of the equivalents of these out of 5052 to save a few bucks, and they failed where the 6061 ones did not! Lesson learned.

From there, we have two gripper types: voxel grippers/feet, and payload grippers. The feet, which I usually refer to as grippers, despite the ambiguity, are what cap the ends of the MILAbot and are for locomoting over the voxel structure. The grippers are what make or break the robot, and throughout this PhD, I have probably not spent enough time with them relative to other subsystems. The grippers are controlled through either a XIAO RP2040 based board, or an ATTiny1624 based board, and feature a similar control interface to the actuator modules.

The voxel grippers in the MILAbot v2 are designed primarily to locomote over a 2-wide voxel profile, though if necessary, can be rotated orthogonally to locomote over a 1-wide voxel profile. This is in contrast with the MILAbot v1 grippers, which necessitate a 2x2 voxel space for a successful step. This does, however, tradeoff some rigidity, and in practice, we do not often need the narrowness of the gripper.

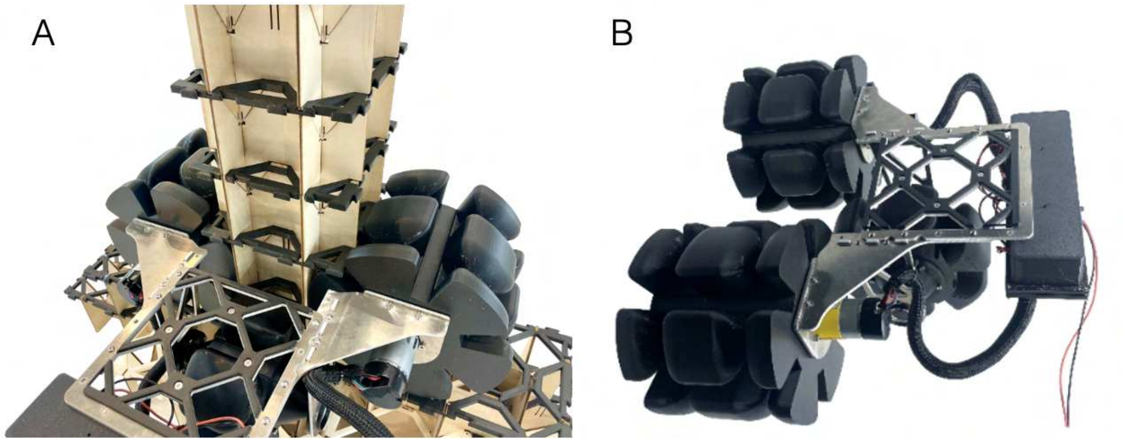

Figure 6-16: A) View of the Elevator robot installed onto a voxel stack. B) Image of the robot showing the two driven wheels.

The mechanism in the gripper is based on latching features, or ”toes” which are driven using a hobby servo with a rack and pinion to extend/retract under voxel beams. This mechanism slides on MGN7 rail (from BulkMan3D) and the toes are cut from 1.5 mm thick aluminum, with a slight chamfer to permit z-axis error correction and tolerance. The base of the foot consists of pyramidal features designed to index into the top surface of the voxel. The size of these alignment cones permits $\pm$ 25 mm of x-y placement error.

The payload gripper for the MILAbot v2, which could be duplicated to increase carrying capacity, consists of a simple hobby-servo based mechanism which latches into the central negative space of a 2x2 voxel array. This gripper is attached to the passive spacers between the main robot body and the grippers and is crained away such that the payload gripper is aligned with 3.5 voxel spaces behind the foot gripper. This allows payload gripping at a variety of voxel block orientations (if you imagine a 4x2xN block, there are 2 available gripping points and 3 accessible gripping orientations).

Finally, as the robot receives commands over WiFi, a robot must have a central, Wi-Fi enabled board. We use the same board as in the MILAbot v1, which is built around a Seeed Studio XIAO ESP32C3 board.

6.2.3 Elevator Robot

To improve the throughput of the system, as discussed in Chapters 4 and 5, we have developed an Elevator robot for feeding voxels to the MILAbot over vertical distances. As the MILAbot needs stairs to traverse up and down, the existence of an Elevator robot enables the assembly of towers and walls with flat edges. The Elevator is designed to operate like a rack and pinion gear system where the underlying voxels act like the rack. Two large wheels, designed to index into the voxel negative space underneath, are driven using geared DC motors to allow the robot to traverse up and down. The Elevator can support significant

Figure 6-17: The planning software, which ingests a mesh, voxelizes it, groups it into structurally stable and assemble-able blocks, sequences the planning, and simulates/controls the robotic assembly. Image from Alessio Zazo.

loads (including an entire MILAbot); the limiting factor in transport is the lateral bending load that this exerts on the underlying voxels, which is supported primarily through the snap-joints, the weakest interface in the system.

6.2.4 Planning

For the software, Alessio Zazo developed a physics-aware digital twin and planning platform for robotic voxel assembly that is summarized here. The software models the stability and kinematics of the MILAbot v2 robot and voxel structures, then uses reinforcement learning to generate build sequences that maintain structural stability throughout construction, accounting for the robot's own weight as it places blocks. It includes a deterministic algorithm that modifies unbuildable geometries into buildable ones. The entire pipeline runs in a browser-based platform that voxelizes meshes, verifies physics, plans and simulates build sequences, and can directly command the physical robot to execute builds. This is diagrammatically shown in Figure 6-17.

6.3 Results

6.3.1 Mechanical Performance

We have characterized the performance of smaller sub-assemblies of the compounded voxel, as well as individual component parts so that we can better extrapolate out the performance of larger scale structures. The compounded voxel block system behaves differently under different loading conditions and assemblies, as the joint system introduces considerable anisotropy. Although this introduces additional complexity in modeling the system, it does not necessarily substantially limit the application of these blocks, as a real structure will not be subject to arbitrary loads in all directions, and we have many levers with which to modify the performance of the blocks or their placement in a structure.

The compression behavior of the block system is very good, as we would expect, as it generally loads the joints along the axis where they have the most load bearing capacity. These results are summarized in Table 6.1 for the tested materials, which are 3D printed PLA and PHA, in a truss-style face and in a plate-style face, as well as 3.2 mm thick 3-ply baltic birch plywood and 1 mm thick 5052 aluminum. The 3.2 mm plywood voxel achieves a compressive modulus of 100.9 MPa at a density of 80.6 $kgCO_2e/kg$, which is about five times higher modulus at the same density as the 3D printed PLA octet system from Chapter 5. This improvement comes from both the switch to plywood as well as the use of a solid face. We can see this impact more clearly if we compare between equivalent materials, as shown in Figure 6-18. The switch from the monolithic 3D printed voxel to the assembled design results in a lighter structure with similar stiffness. We see minor improvements in specific modulus between these two generations. The switch from truss to solid plate introduces much more improvement to the specific modulus: a bit over two times higher specific modulus can be seen between the solid plate face PLA versus the hollowed out truss face PLA voxel blocks.

Material	Modulus (MPa)	Density (kg/m^3)	Specific Modulus ($MPa/(kg/m^3)$)
PHA truss	27.5	66.4	0.4
PLA truss	13.7	67.9	0.2
PLA plate	39.4	88.9	0.4
Plywood 3.2 mm	100.9	80.6	1.3
5051 aluminum 1 mm	39.8	100.0	0.4

Table 6.1: Material properties comparison

The tensile and shear behavior of the blocks is primarily determined by the snap fit joints. The strength and stiffness of these joints can be modulated through the design of the snap fit, though the need to keep the voxels robotically assembled introduces limits on the installation force of the blocks, which must be less than the force exerted by the weight of the robot, as well as the extension of the snap fit from the body, which must stay within the alignment cones to facilitate assembly. This is also constrained by the need or desire to keep the system reconfigurable. While we could strengthen and stiffen the snap fits more than what is demonstrated here, it would be difficult to do so and still meet these requirements.

Because in tension the blocks primarily load onto the snap fit joints, the structure is much more compliant, weaker, and fails at lower amounts of deflection. The plywood blocks have a tensile modulus of 5.7 MPa, and a single snap fit supports a peak load of 40 N. The plywood snap fits cannot support multiple load-unload cycles without failing, however, the friction of the system is fairly high, so even after complete failure of the overhanging snap element, each joint supports up to 7.8 N before complete disengagement. The 1 mm thick 5052 aluminum is designed with a stronger snap fit that supports a peak load of 68 N per snap fit joint. However, the joint is poorly designed in that it enters the plastic deformation

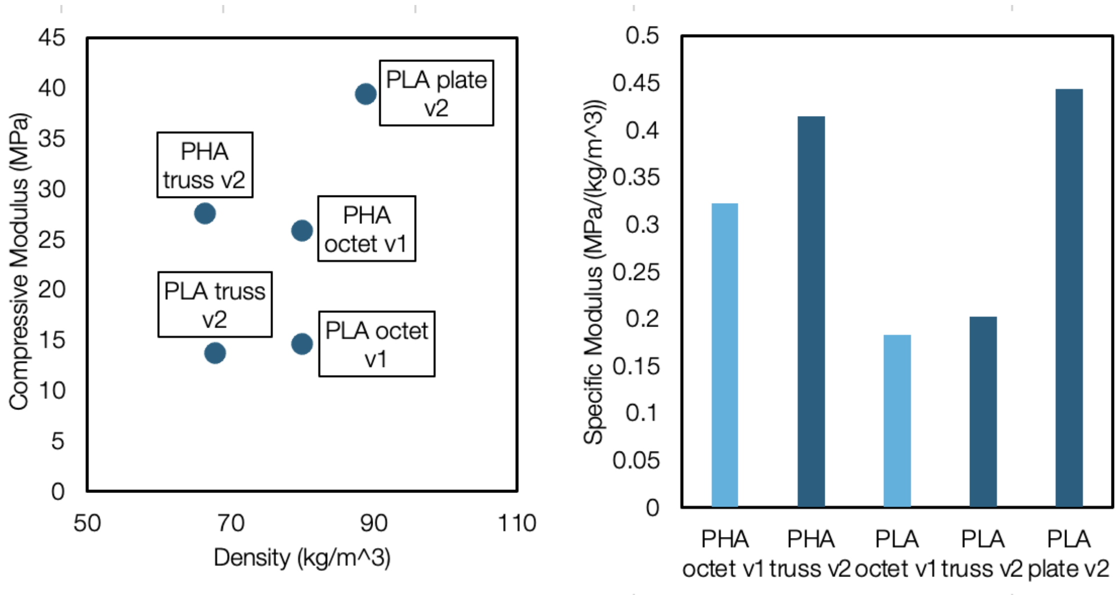

Figure 6-18: (Left) Compressive modulus vs density for the 3D printed PLA and PHA voxels from this chapter as well as the earlier design. (Right) The specific modulus of these voxels, with the prior monolithically printed voxels in light blue and the assembled voxels from this chapter in dark blue.

region of the snap fit, and upon deinstallation, will deform the joint such that there is almost no holding force. This is because it was dimensioned for 6061 instead of 5052. The joints can can be manually fixed, but future revisions should take care to keep the system within the elastic regime.

Allowing the blocks to carry much less tensile load keeps the structures reconfigurable. By designing in alternate joint systems, we can still enable good load transfer to support bending loads. The dove tail joints along the top or bottom surfaces allow tensile loads to be transmitted along those planes. Staggering the blocks in a brick-wall style layout then allows shear loads to be transmitted through the snap-fit joints sufficiently to engage the dovetails. Under bending, the dove tail joints carry the tensile load while the opposite side jams to support a compressive load.

We can use the strength of these joints to estimate the bending capacity of beams made from these compounded voxel blocks. Although the dove tails we used are primarily 3D-printed PLA, typically at 15% infill, we can fabricate these out of a range of materials and introduce stiffeners into the top voxel part as done for some of the aluminum voxels to modify the performance of these joints. Single dovetails were tested in a range of materials (results are summarized in Table 6.2). The basic printed PLA joint supports a peak load of 583 N, the 1.6 mm aluminum version support 1117 N, the steel version support 1654 N, and the plywood versions support 151 N and 257 N depending on orientation relative to the external grain. Because the top design is by default 6 mm thick (although this can be modified), this means we can increase the thickness of any of these dovetails up to 6 mm without modifying the robot grippers.

Test Type	PLA solid infill 6 mm	Aluminum 6061 1.6 mm	Mild steel 3.2 mm	Plywood parallel grain 3.2 mm	Plywood orthogonal grain 3.2 mm
Single joint (N)	583	1117	1654	151	257
2-wide, 1/2 load (N)	1166	2234	3308	302	514

Table 6.2: Joint strength comparison

Based on these results, we can estimate the dimensions of beams made from voxels with different dovetail base materials under bending loads. For a residential application, we consider a 4 m span under a 20 p.s.f and 100 p.s.f load based on the IBC regulations for an unoccupied and occupied roof. We use half the ultimate load for the dovetail, as summarized for each material in Table 6.2, for a 2-voxel wide beam and increase the height of the beam's cross section until it can span the 4 m gap. For the 20 p.s.f roof load, using the PLA dovetail, we get to 3.5 x 2 voxel cross section or is 26.25 cm x 15 cm, which is practical and workable set of dimensions. For a 100 p.s.f load instead, this would require a 17 x 2 voxel cross section (127.5 cm x 15 cm), which becomes impractical in the amount of thickness it contributes. Switching to a 3.2 mm thick aluminum dove tail would allow us to decrease that height to a 8.5 x 2 voxel cross section or 63.75 cm x 15 cm, which is half that of the PLA system. A 6 mm aluminum dove tail would let us drop to a 4.5 x 2 voxel cross section, which is practical in terms of overall dimensions, assembly time, and material use. If we further increase our span to 10 m, toward commercial applications, with a 6 mm aluminum dove tail, we would need 26 x 2 voxel cross section (195 cm x 15 cm). This, again, is likely too tall of a cross section for practical deployability, and would suggest that we would need to revisit the voxel design for this application, and likely additionally consider shaping the cross section, such as in a I-beam style shape, or as a truss, for better performance.

The flexural stiffness of the system is also relevant for determining appropriate dimensions and designs for the voxel system. We can first consider the performance of a beam composed only of standard voxels, which only has the dovetails along the top surface. For a voxel beam to stay assembled, we require a minimum height of two voxels, to introduce the brick stagger. This means that under bending, although the bottom layer of voxels will separate, the top surface of these acts as a constraint on the next level of blocks. We can see this with a 3-point bending test on a 2-voxel tall, 2-voxel wide beam with a 5-voxel spacing between the supports made from plywood voxels with PLA dovetails. The structural stiffness of the beam is 0.48 kN/mm, which corresponds to a flexural modulus of 10.3 MPa for an idealized version of the beam, which is substantially lower than its compressive modulus. If we instead consider the cross-section of beam to be 1-voxel tall, as this corresponds to the structurally active layer of voxels, converting from the structural stiffness to the flexural modulus results in 100 MPa, equivalent to the compressive behavior. This means that we

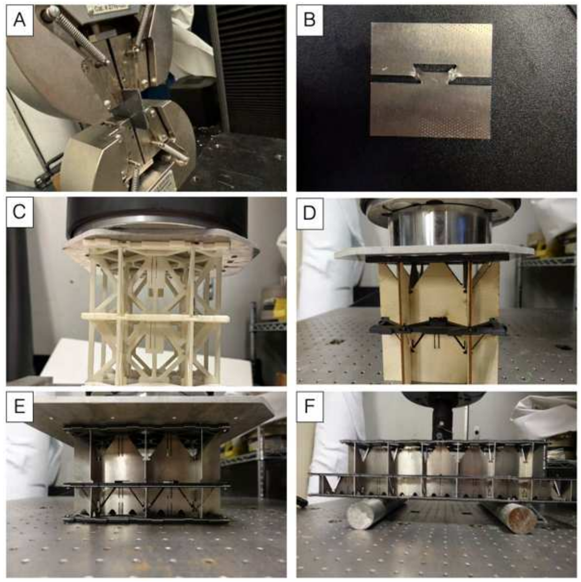

Figure 6-19: Examples of mechanical testing set ups. A) and B) individual dovetail samples. C) D) E) Compression testing examples for PHA, plywood, and aluminum. F) 3-point bend test set up.

can build up voxel beams with an additional base layer of voxels to provide bending stiffness.

We can alternatively use the voxel bases that have dovetails on the bottom layer as well. This means that we do not need to waste the material and space of building out an additional layer of voxel blocks. We repeat the same 3-point bending test but using 1 mm thick 5052 aluminum voxels with the bottom layer made from the half-height double sided voxel bases with aluminum stiffeners in the top faces. This gives a structural stiffness of 0.51 kN/mm, which corresponds to a flexural modulus of 33.7 MPa, which is 84% the value of the measure compressive modulus, which indicates reasonable approximation of a continuum version of the beam.

These are relatively short beams and the presence of additional joints could introduce additional compliance or slop. To evaluate this, we fabricated a longer beam from the standard plywood voxel with a 2-voxel by 2-voxel cross-section and a 25-voxel long (1.875 m) spacing between the supports. The supports are also voxels, building out a small bridging structure. To approximate a bending test, we measure the displacement at the midpoint with calipers with increasingly heavy cylindrical weights placed on top. We then back out the stiffness from that. This results in a 9.7 MPa flexural modulus, which is 94% the value of the shorter beam. This suggests that we can extrapolate out the stiffness/modulus values from the smaller scale tests.

Using this beam, we can test another method for improving the bending behavior: post-tensioning. We fabricate two 1.6 aluminum plates that hang onto the ends of the beam that we fasten two M5 threaded rods through so that they run along the base of the voxels. For this test, I first inverted and loaded the beam with 4 kg of weight and tightened the rods to

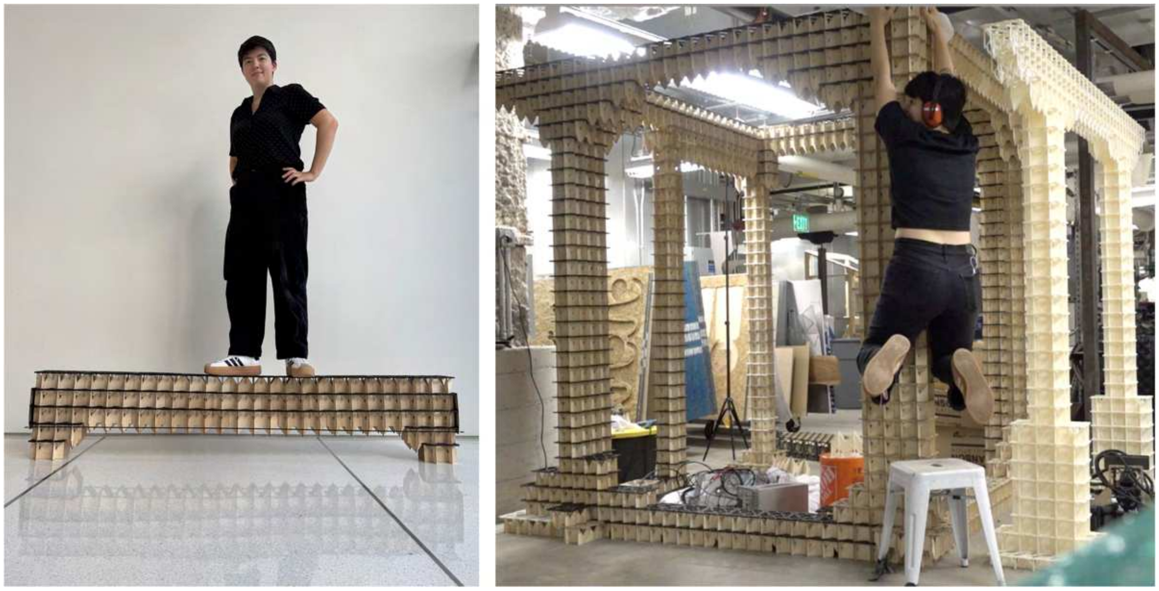

Figure 6-20: (Left) Post tensioned beam with human load. (Right) Testing the tiny house.

finger-tight. The beam is then unloaded and placed right-side up. The measured structural stiffness of this beam is 60.2 MPa, which is about 60% that of the compressive modulus, while increasing the weight of the structure by 13%. This improvement is with a fairly approximate methodology, and we could likely improve the performance. The beam, with an additional voxel layer, is shown in Figure 6-20. Post-tensioning is a relatively simple operation, which could be replicated to improve the tensile performance of the beams as well.

Using the results of the post-tensioned beam, we can evaluate its suitability for a building application. For this, we consider a 5 m span, similar to Chapter 5, and consider both the equivalent loading scenario from Chapter 5 (20 p.s.f) and the higher 100 p.s.f loading scenario for an occupied roof. Using a simplified beam model, for a simply supported beam, we calculate the required height of the voxel beam cross section to meet L/360 maximum deflection criteria. For the 20 p.s.f load, we meet this at a 13 x 2 voxel cross section, and for the 100 p.s.f load condition, we meet it at a 21 x 2 voxel cross section. This is consistent with the results for the expanded Miura voxel system from [139] analyzed in Chapter 5, which has similar modulus and density values to this system. However, the material blend for this system (plywood, PLA, and steel) has lower embodied carbon than the aluminum sandwich composite used in that system.

Overall, the voxel block system described here performs competitively with other architected lattice systems and outperforms other robotically assembled versions. Figure 6-21 plots the compressive modulus vs density for selected discrete lattice systems as well as continuously manufactured microlattices from the literature. This shows that the voxels in this chapter (and more broadly in this thesis) are competitive with other lattices, though because we can support robotic assembly, are better suited to scale to larger structures.

If we consider the specific modulus of these systems, we can see that the highest performing continuous microlattice systems from [184] still outperform the highest performing discrete lattice systems from [46] and [72] (at least if we use specific modulus as the comparison metric). This suggests that voxel systems do still pay a penalty from the constraints of joint systems, part manufacturing, and overall assembly processes. Adding robotic assembly as a requirement introduces further constraints, and the system demonstrated here outperforms the next highest performing robotically assembled voxel system [71] by an order of magnitude.

6.3.2 Basic Assembly

The MILAbot v2 performs a similar range of motions to the MILAbot v1 at slightly faster speeds. Overall assembly processes are similar, but for the MILAbot v2, we permit the loading of voxel blocks in alternate orientations to reduce the number of assembly movement types.

2D to Voxel Block

We go from 2D faces to voxel blocks using a gripper end-effector, an assembly jig, and part feeders. These are shown in Figure 6-23. The gripper end effector is designed with the same mechanical interface as the top part of the voxels, which allows the robot to pick it up. It is powered by an on-board 5V USB battery and controlled with an ESP32C3 Seeed XIAO, which is WiFi enabled, allowing us to control the gripper from the same interface as the robot. The gripper is a simply designed hobby servo driven pincer which can pick up the flat 2D faces and insert them in the assembly jig (designed by undergraduate researcher Ryan Hoffman), which uses flexural arms to help guide and stabilize the part. After placing the body faces, the robot can drop off the gripper onto the underlying voxels and pick up the top part using its standard grippers. This is then dropped on top of the 2D faces and pressed/pushed into place. The block can then be removed. These steps are shown in Figure 6-24. Overall, the robot takes about 5 minutes per 2x2x1 voxel. This is volumetrically faster than manual assembly of many prior voxel assembly systems, such as [91], but it is slower than manual assembly for this design, where I average at about 30 seconds per 2x2x1 block. The speed can be improved through faster operation of the robot (less pauses), parallelized operation, or alternately a dedicated machine system could be used. Even at this rate, assuming we can pre-assemble some voxels, this would not be a limiting factor.

Blocks to Structures

The MILAbot assembles blocks into structures by carrying a voxel block to a new position, dropping it into place, and stepping on it to engage the snap fits. Depending on the stagger of the block, the blocks can be tiled into almost arbitrary voxelized geometries, with the

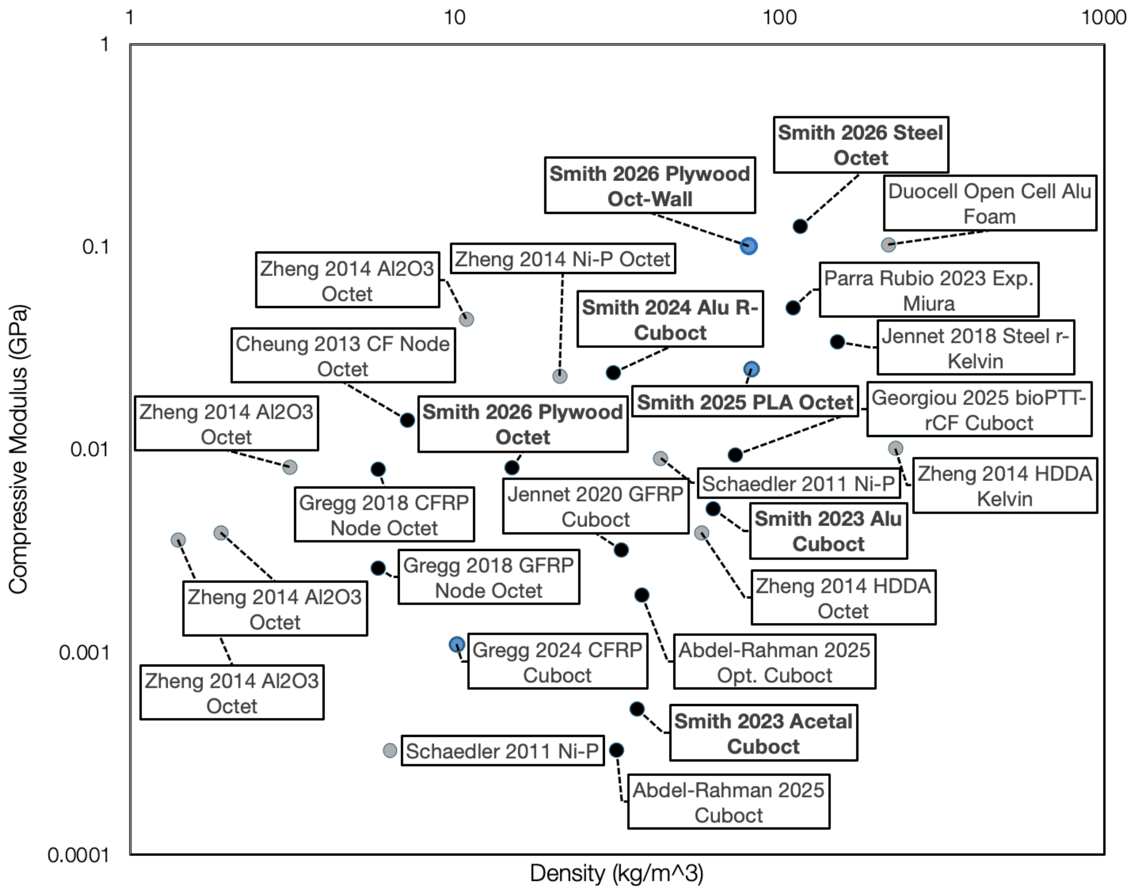

Figure 6-21: Compressive modulus (GPa) vs density (kgCO_2e/kg) for selected discrete (black dots) and continuous microlattices (grey dots). Systems which have been used for robotic assembly are blue dots. Voxel systems from this thesis are bolded. The compounded voxel block from this chapter demonstrates the highest modulus of a robotically assembled system at a density competitive with manually assembled or continuously fabricated approaches.

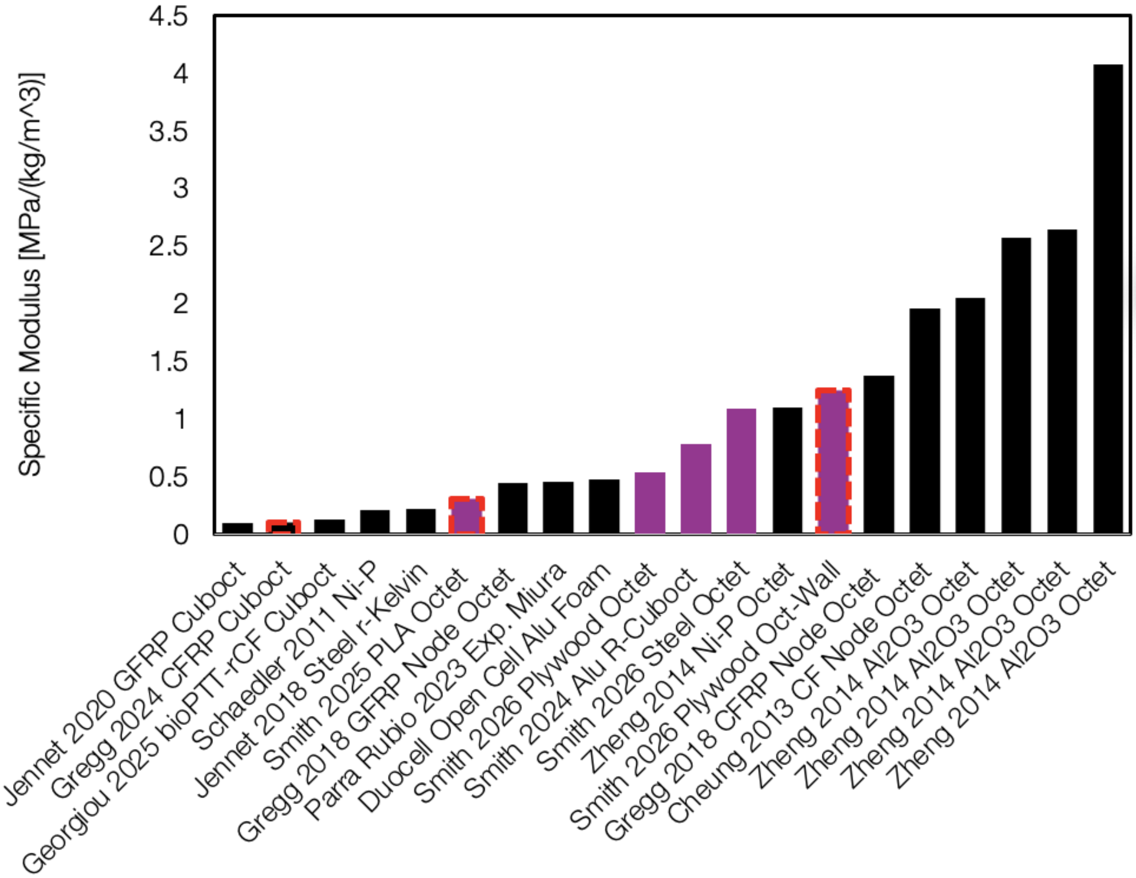

Figure 6-22: The specific modulus of selected architected lattices, both voxel and continuously fabricated. Works from this thesis are in purple, and red dashed lines indicate robotic assembly. The system in this chapter demonstrates order of magnitude higher specific modulus than the next robotically assembled system.

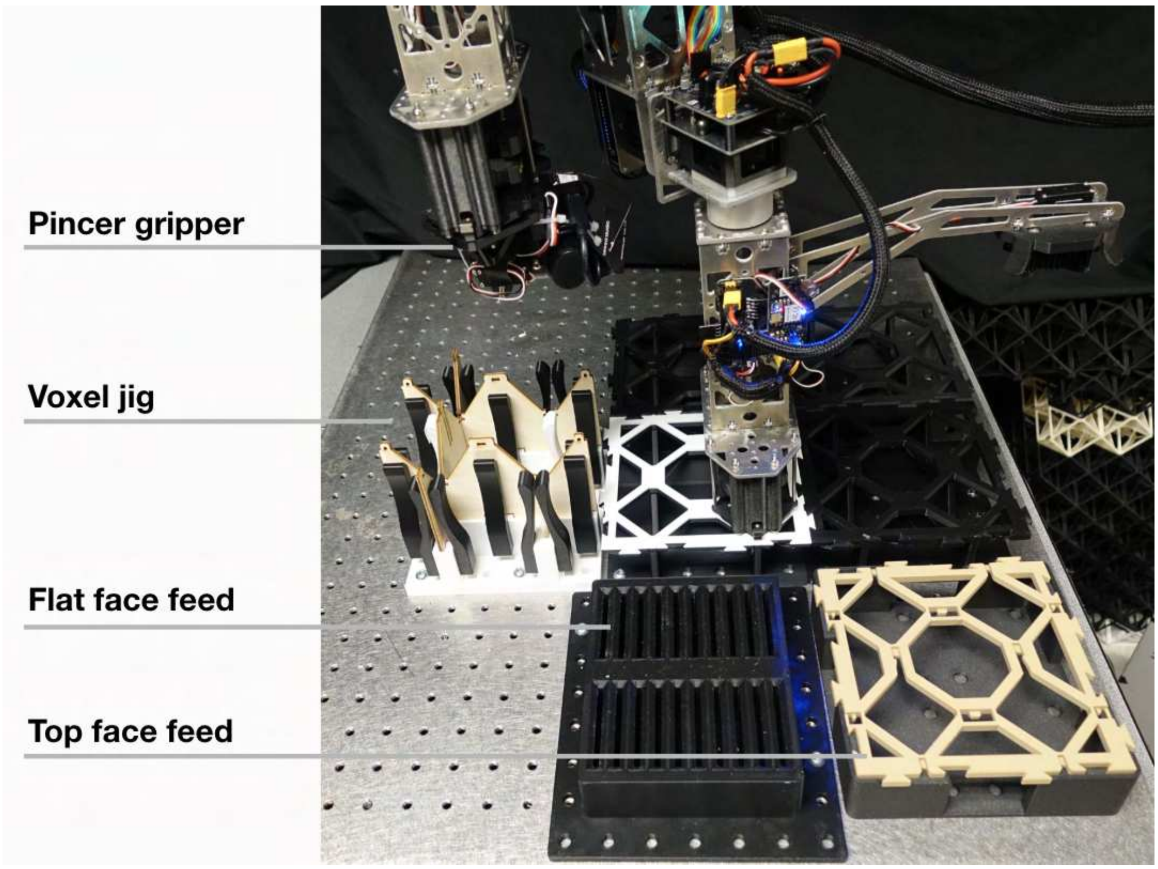

Figure 6-23: 2D faces to voxel block assembly station.

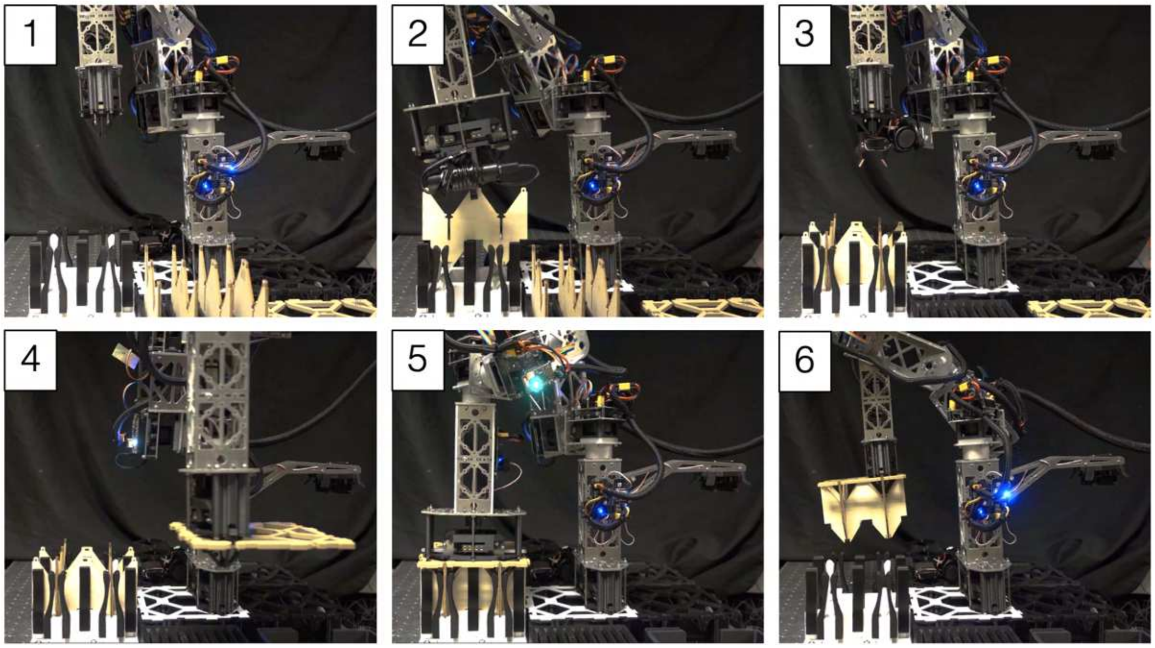

Figure 6-24: Stills from the 2D to block assembly process

greatest design limitation introduced by the accessible space of the robot itself. For example, the steepness of stairs is limited by what the robot can reach.

The most basic, fundamental operation for the MILAbot is building out a base or first layer. Starting from a set of "seed voxels" which at a minimum need to be 3x2 voxels in area, with a step-down stagger, the MILAbot can extend out the structure. Lines are achieved via block staggers along a single axis, while planes or joints are achieved through a two-axis stagger. An example line assembly is shown in Figure 6-25.

The MILAbots are generally run so that they start from separate voxel feeds and can assemble outwards to join their structures. An example of this is shown in Figure 6-26, where we operate both robots at once to build out half of the first layer of the prototype tiny house. This example also points to a current limitation of the system: we do not a clean mechanism for fixing the first layer down to the ground, so in practice we add additional flanges off of the seed voxels that weights can be placed into to stabilize the structure as it is being built. Even so, this build still resulted in a small gap at the seam, which the robots were able to close as it fell within the alignment cones of the voxels, and because the structure is small enough for a single robot to deform. For larger structures, with larger seams, this is likely to become a critical failure point.

Once a first layer is established, the robots can extend the structure by placing single layer, double layer, or additional staggered blocks. These can and should be interleaved for additional stability. Figure 6-27 shows an example build of the filetted features at the base of the tiny house structure, where two robots place staggered blocks to form an interleaved seam.

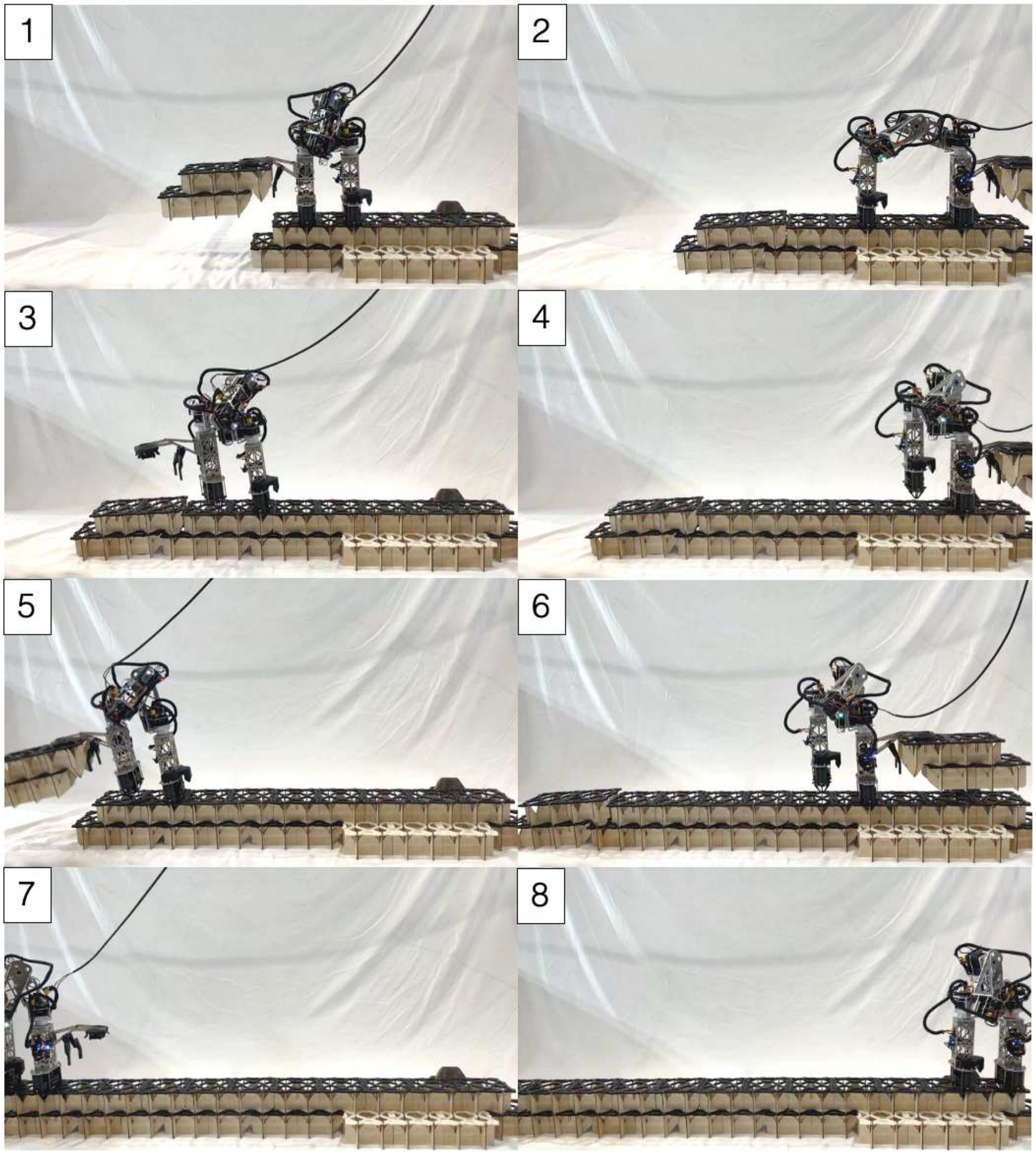

Figure 6-25: Assembling a line on the ground as a base layer.

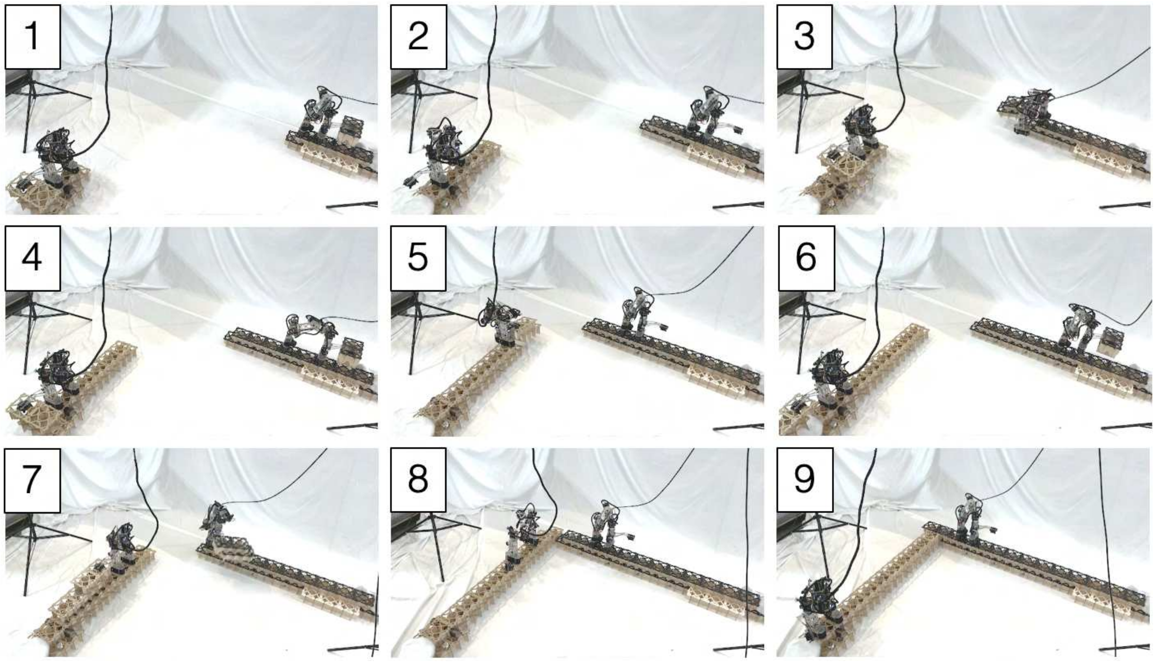

Figure 6-26: Two robots assembling a base layer

The process of assembling a first or base layer is akin to that of doing overhanging or cantilevered features. The voxels are able to support relatively long cantilevers as the dovetail joints are loaded in tension by the robot weight. The limiting factor is the torque applied to the joints at the base of the cantilever or in the corbeled featured, as this loads the snap fit joints. An example build of a cantilever is shown in Figure 6-28. Here, the robot creates a 9-voxel long (or 675 mm) overhang, which is the limit before the snap fit joints started pulling out. This amount of overhang means that two robots in concert can create a 1.35 m span, which can be extended through increasing the size of the corbel. This amount of span is sufficient for a wide range of applications, for example, it exceeds the distance required for standard stud framing spacing.

However, this span is likely still insufficient for many applications, as we likely do not want to accept the need for a column support every 1.35 m. To extend this, we have several options. The first, as mentioned, is to rely on the corbel to steadily build out a supported overhang, though this will also run into limitations and is not highly material efficient. The next option is to develop robot maneuvers which more thoughtfully load the structure. That is, current maneuvers swing out the voxel payload from the structure, creating dynamic moment arms that are the cause of structural failure. It is likely possible to extend the overhang range via inline and gentle movements. The last option, which holds the greatest promise toward extending the functionality of the system, is to post tension through the anchor point of the cantilever and the tower that it rests on. For a real structure, we would likely want to do this anyways, as e.g. wind loads on a roof could disassemble the voxels. This would eliminate the failure mode and likely push it to the failure load of the dovetails instead.

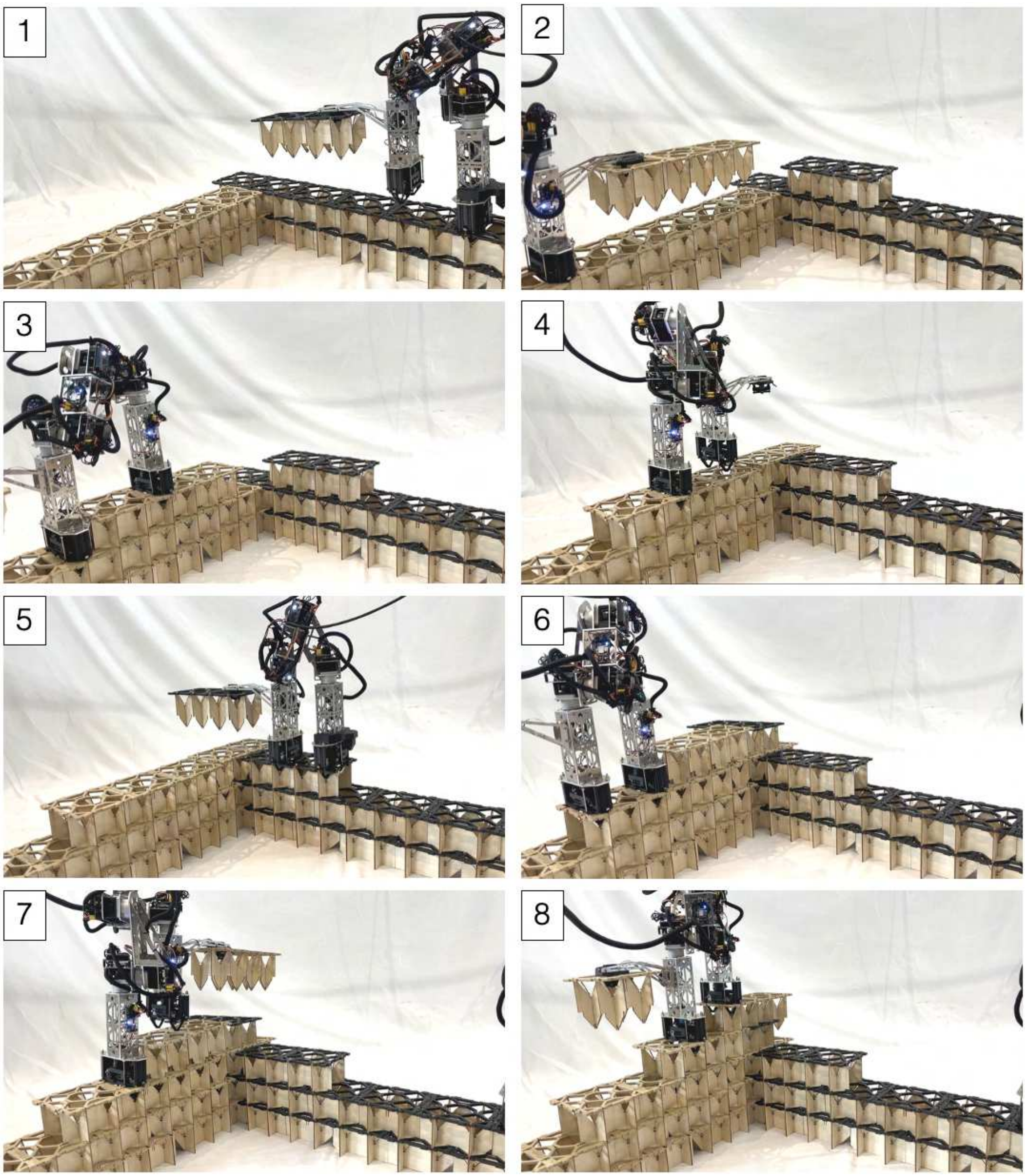

Figure 6-27: Assembling a staircase

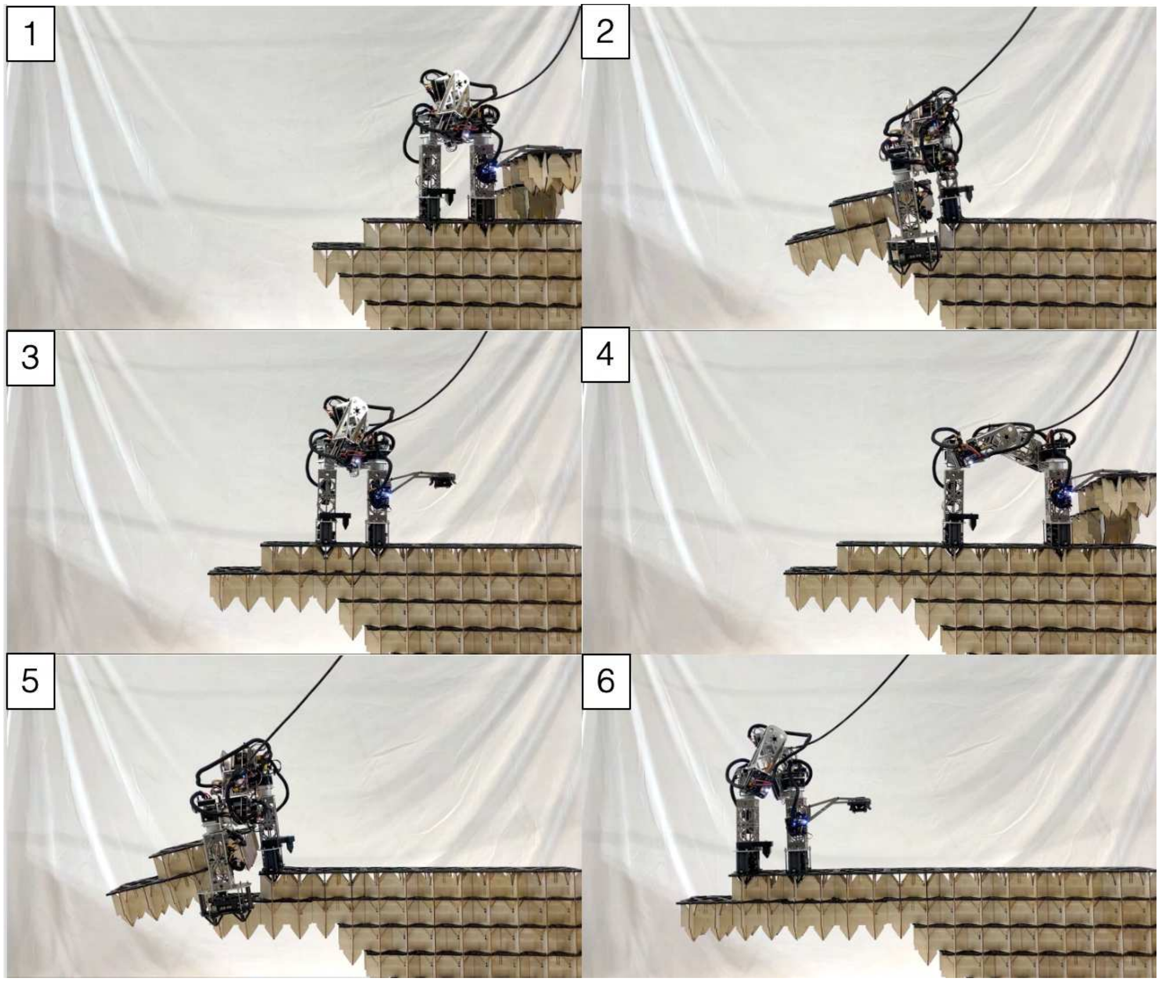

Figure 6-28: Assembling an overhang.

6.3.3 Demonstration Structures

Prototype House

The robotic assembly system we have developed holds promise for architectural scale applications. We have demonstrated a route toward fast, cost effective, and sustainable construction using low cost robots and almost arbitrary materials. While this is motivated by the general challenges of large-scale digital fabriction, much of the work of this chapter is also motivated by a specific collaboration we have with Druk Holding and Investments (DHI), Bhutan's sovereign wealth fund, around sustainable robotic construction. From DHI: "This initiative supports the vision of the Gelephu Mindfulness City (GMC), and advances DHI's 10X Roadmap to catalyze national transformation through technology, talent and global partnerships. The project will be operated through the Multi-Channel Innovation Fund (MCIF), a key initiative under DHI's Innovation Strategy to establish innovation as a pillar of Bhutan's economy. The agreement, spearheaded by the Jigme Namgyel Wangchuck Super Fablab (JNWSFL) within DHI's InnoTech Department, will explore advanced robotics and engineering, blending Bhutanese traditional architecture with modern design and fabrication technologies. By collaborating closely with MIT researchers, this initiative aims to position Bhutan and GMC at the forefront of sustainable and intelligent construction systems."

The goal of this project is to autonomously deploy architecture-scale structures made using locally produced materials and local FabLab produced robots. To this end, we have built a 2 m x 2 m x 2 m prototype tiny house as a small, though at scale, construction artifact. This structure is made from predominantly plywood-based voxel blocks, using either PLA or PHA (as developed by Crafting Plastics!/Vlasta Kubusova) for the top parts. It additionally features a decorative, breeze-block style PHA window.

Along with the base structure, we fabricated an archway and a load bearing bench as additional artifacts for the FAB26 conference held at MIT, where people were free to use/disassemble the bench and engage with the other structures. These structures were assembled manually from some robotically assemble sub-components, and this points towards another potential deployment method where we might prefabricate e.g. wall panels, roof joists, etc. robotically, but still assemble these larger elements together manually, which is a fairly easy process as the block system is lightweight and error correcting. Through this assembly and hands-on design process, the tiny house has been assembled, disassembled, and reassembled into its component parts at least twice for the entire structure, and more for some of the sub-assemblies from testing the robots. This also points toward the reconfigurability of the structure as a real feature that could open doors toward different types and functions of structures.

The tiny house is also partially clad in a bioplastic cellular skin system designed with a Bhutanese geometric pattern. This was developed by Crafting Plastics!/Vlasta Kubusova for this installation, and while in this iteration it functions primarily as an aesthetic feature, it points towards another significant area of future work considering how we enclose or skin voxel structures.

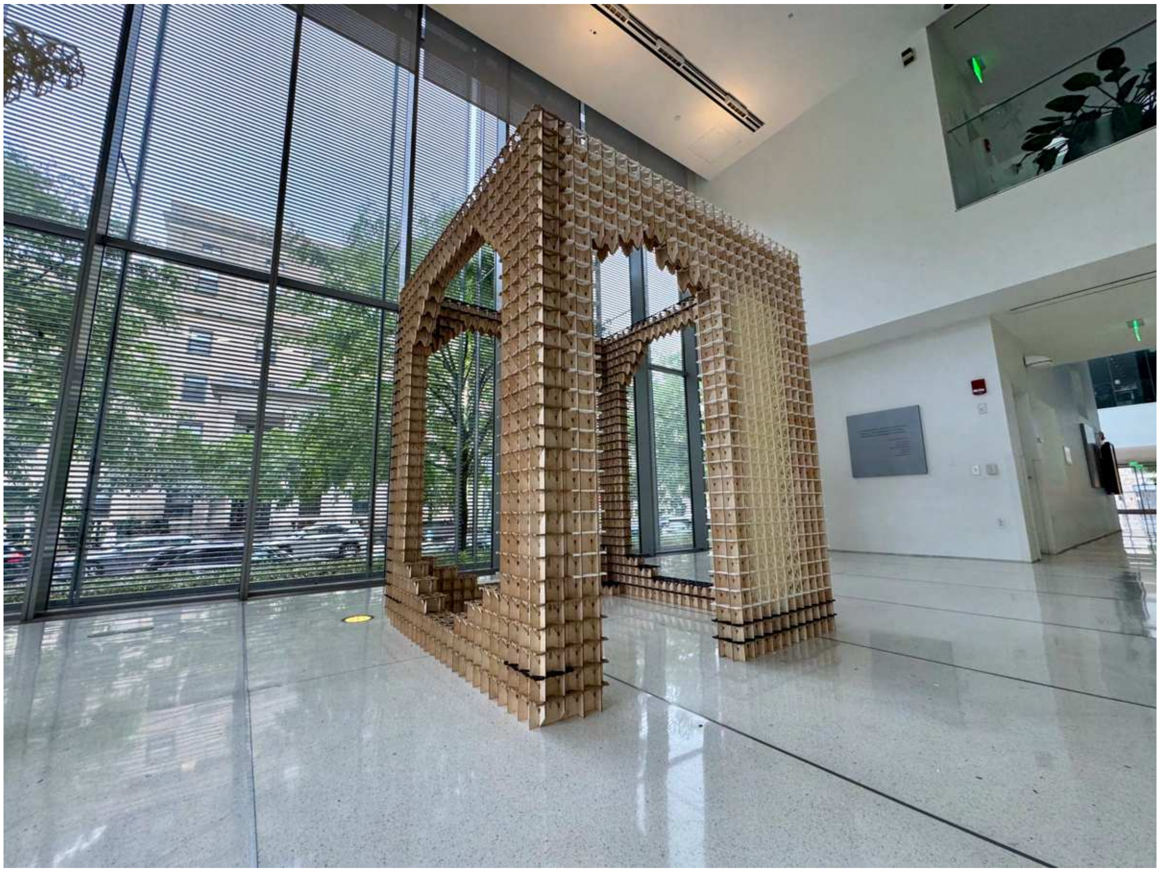

Figure 6-29: Tiny house voxel framing.

Figure 6-30: A) Detail view of bioplastic cladding with decorative Bhutanese inspired geometric design. B) Front of the the tiny house with example cellulur cladding. C) Archway and bench next to house. Photos from Vlasta Kubusova.

Voxel Vehicle

With Toyota, we have also been exploring using voxel systems for reconfigurable mobility. Specifically, we are working on a voxelized concept of the Kids Mobility vehicle, which is a child sized demo vehicle. We are developing this around the idea of a vehicle whose shape can be endlessly reconfigured as demands change. Starting with the Kids Mobility concept, we can imagine a car that grows as the child grows. Expanding from that, we might envision a sedan that could morph between a two-seater or a 4-seater, or an SUV which adds and removes vehicle footprint as appropriate for each trip. These are all very conceptual, but we are starting with some basic proof-of-concept demos. We have produced, using the same building system as the Bhutanese house, two iterations of reconfigurable mini-cars. These are sized based on the dimensions of a Kids Mobility vehicle for a child in the 7-9 year age range, which roughly corresponded to the size of wheels and axles more commonly marketed for go-karts, so the vehicle design was revisited to support this external hardware. We used these cars as an opportunity to explore manual fabrication for these systems. With the aluminum car, I was able to snap together the chassis in under 2.5 minutes. This is, for better or worse, much faster than the robot. The chassis was then disassembled (7.5 minutes), reassembled, disassembled, and reassembled again. The plywwood car was then also built, as the three-point bend testing had revealed essentially equivalent load capacity as the aluminum version.

6.4 Future Work and Limitations

The goal of this work is to reach closer toward a future where we can sustainably build high-performance structures using locally available materials. So far, we have demonstrated in a controlled, lab environment the assembly of full-scale, though small and incomplete, versions of relevant structures, such as temporary buildings or furniture. These demonstrations, and the preceding comparative studies, indicate the potential for this kind of building system for large-scale applications, but they largely do not verify or validate this potential.

Our immediate next steps are around outdoor, in-situ testing of the assembly system in Bhutan, where we will assemble a structure similar in scale to the tiny house and evaluate its longer term, outdoor performance. This will likely be targeted toward building e.g. a bus stop, or other small-scale public infrastructure, such as seating for parks or pavilions for outdoor market places. This will help us evaluate the performance of the structure in a realistic environment over a longer period of time.

However, this will still leave many important performance metrics unevaluated. Our goal with this system is to be able to eventually build actual buildings and there is a long way to go to that. Some example concept art from Alessio Zazo and Amira Abdel-Rahman is shown in Figure 6-35. This will require evaluating and understanding the voxel behavior under larger loads, its fire performance, thermal or hygroscopic behaviors, and more. It will also require developing insulation, sealing, and cladding methods for the voxel system, as well as potentially better methods for interpolating slopes and curves.

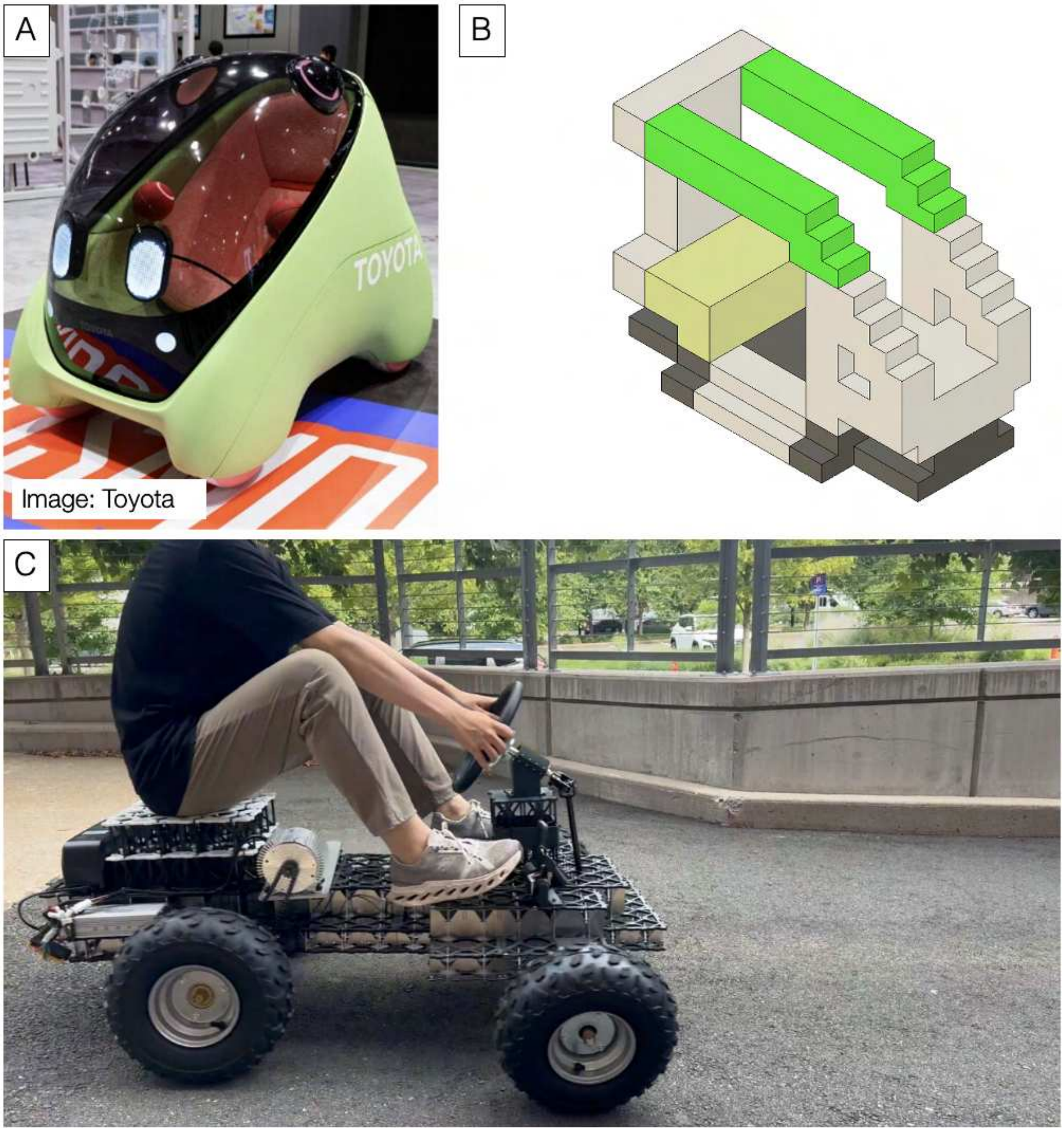

Figure 6-31: A) Kids mobility vehicle concept from Toyota. B) Coarsely voxelized version. C) Assembled main chassis under operation.

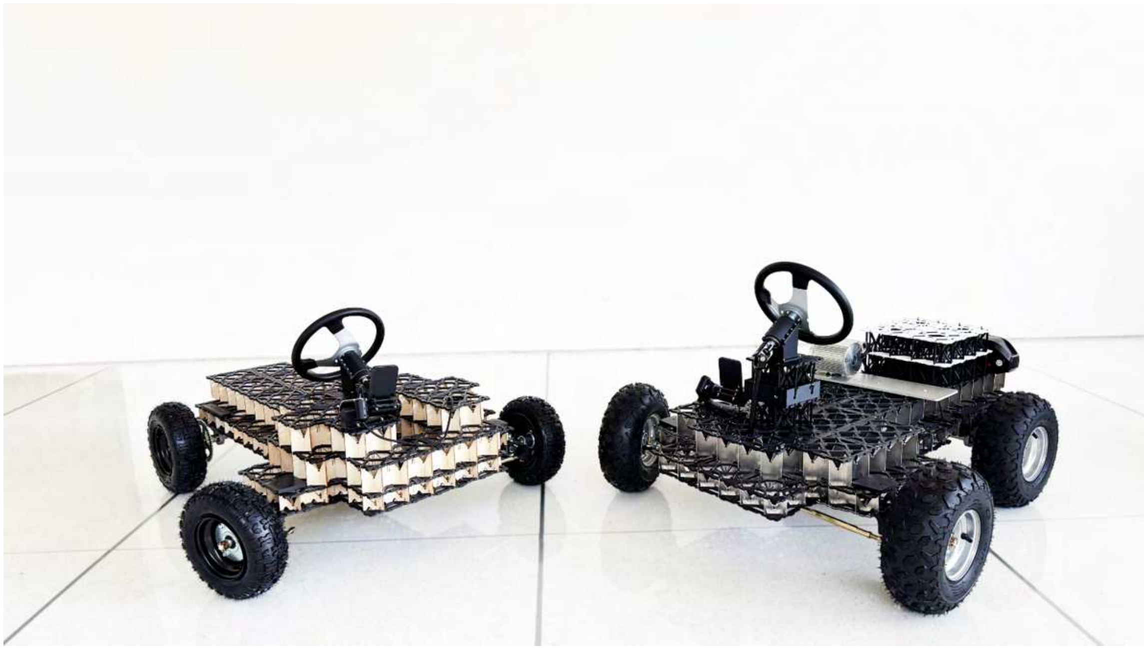

Figure 6-32: The two assembled mini-voxel cars.

Figure 6-33: (Left) Aluminum voxel car. (Right) Wood voxel car with model (image: Vlasta Kubusova).

Figure 6-34: Concept art of temporary voxel structures. Background images from Bhutan GMC.

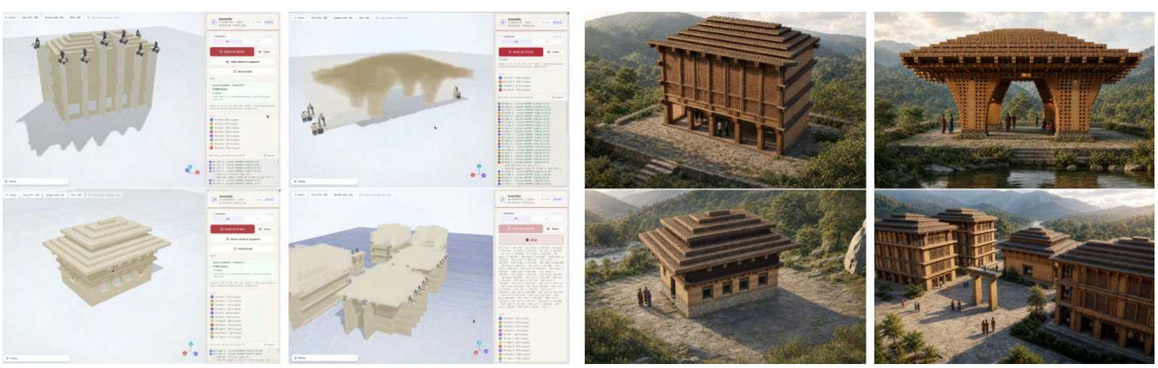

Figure 6-35: Building-scale voxel assembly concepts. (Left) Planning and simulation. (Right) Rendered concept art. Images by Alessio Zazo and Amira Abdel-Rahman.

In the nearer term, there are many immediate nexts with this robot material system.

- Post-tensioning: We have prototyped post-tensioning a voxel structure and it holds promise as a reversible method to constrain final degrees of freedom in the structure without adding significant mass or labor. It also opens questions about whether this can or should be automated by the robot, and whether or not we can do more intelligent routing of tendons throughout a structure for improved performance.

- Robot feeds and heterogenous swarms: We still manually feed in voxels to the system and this is a significant bottleneck. Additionally, we can consider the prefabricated assembly of larger subcomponents and how the MILAbots, or a different morphology of robot, might be able to work these.

- Anchoring and seams: While we can build out a first layer, we do not yet have automated methods for anchoring to a foundation. We can leave this as a manual task, but it then introduces the very significant problem that the initial anchor points need to be accurate enough for a segregated robot swarm to close the seams of their assembly sections. Alternate approaches may be to introduce compliant seam voxels, metrology for the robots, or planning methods that allow the robots to start closing seams earlier, and progressively.

- Robot planning: There is still much more space to explore in the planning and coordination of these robots, which has not been a focus of this thesis. More physics aware planning (and design), better feedback systems, and planning algorithms that permit (or handle) robot intersections will likely boost the performance of the system.

- Skin systems and insulation: A real building (though not all functional structures) will require a real seal around the structure. Prior CBA work has explored cellular skins, akin to feathers or scales, over voxel structures. This approach, while suitable for creating certain prototype airfoil surfaces, is unlikely to ever provide sufficient sealing in a variety of conditions for real building applications without some form of real sealant. However, there are likely applications where this could still provide benefit. One step larger would be to consider panelized skins that serve to additionally interlock the voxels and provide tensile carrying ability. This once more, is likely to create problem areas around the seams for sealing, but will at least provide structural benefit. One step larger from that still would be to consider continuous membrane options, akin to house wrap, placed directly over the voxels, which could still provide tensile load carrying ability. The more "standard" option would be to simply treat the voxels as framing elements, and add insulation, sheathing, sealing, and cladding fairly normally, as is done in the estimates in Chapter 5.

- Material production: We have created a design pattern which is fairly material and production method agnostic. However, a goal of the project is to produce these voxels from locally sourced, and potentially locally synthesized materials, and that has not yet been realized, and is beyond the scope of this thesis. However, building out this capability would be very exciting toward realizing a highly localized, yet precise, method of building.

Chapter 7

Conclusion

This thesis aims to advance large-scale digital fabrication not by building larger or more precise machines, but by embedding the logic of construction into the building material itself. Just as the shift from analog to digital computation moved reliability out of imperfect hardware and into an error-correcting encoding, this work treats the discrete building block as the carrier of the structural, manufacturing, assembly, and functional logic that large-scale fabrication has conventionally demanded of the machine. By embedding this logic into the material system, we substantially reduce the demands on the robotic systems, enabling assembly by simpler and more parallelizable robots. While prior work considered elements of the voxel's performance in isolation: structural performance without considering assembly, assembly in the absence of structural performance, geometry without manufacturability, and so on, this work considers each of these elements together and seeks to unify their function within the voxel block design, enabling faster, cheaper and more sustainable assembly of large scale structures. This way, the practical application of voxel systems for large scale assembly becomes much more realizable.

Material production across industries contributes about 40% of global greenhouse gas emissions [3]. A practical approach to reducing that is by reducing the amount of material that we use through more structurally efficient building geometries. However, it's often prohibitively expensive to design and build more optimized structures. By developing fabrication methods where cost is decoupled from geometric complexity, as digital fabrication in general promises to do, we can enable a future where structures are much more optimally built, reducing the embodied carbon without sacrificing performance. This thesis advances this goal through the development of these fabrication-aware voxel block systems, which can be cost-effectively assembled into meaningfully load-bearing structures using low-embodied carbon materials.

In the preceding chapters, this principle was developed in several dimensions. Electronics-integrated voxels demonstrated that power, signal, and computation could be routed through the block itself, enabling the direct-write fabrication of electromechanical structures (Chapter 2). We used these blocks to (re)assemble a set of reconfigurable, interactive electronic furniture-scale structures, such as a dance floor and force-sensing bridge, without loss of mechanical or electrical function, demonstrating how we might one day create large-scale electromechanical structures in a simple, building block based approach. These functional blocks were then used to build modular robots capable of assembling further robots, a step toward recursive, self-scaling assembly (Chapter 3). We used a first robot to assemble a second 6 DoF assembler robot and used these to demonstrate collaborative assembly of load-bearing cm-scale primitives, as well as a modular quadcopter and quadruped. To advance in size-scale, hierarchical interlocking voxel architectures and a new modular assembler were introduced to increase assembly throughput and mechanical efficiency while keeping the material system manufacturable from flat feedstock across a range of materials (Chapters 4, 5, and 6). This system improves throughput relative to cost compared to prior voxel assembly system by more than an order of magnitude. Similarly, compared to existing gantry and robotic arm approaches to additive manufacturing, these new assembly systems can exceed their volumetric assembly rate at an order of magnitude lower cost through parallelization. Finally, the system was evaluated at architectural scale, both through a comparative study against existing construction methods and through the assembly of load-bearing structures from reusable parts and renewable materials (Chapters 5 and 6). The comparative study demonstrated that voxel approaches can substantially reduce the embodied carbon associated with building a given structure, especially compared to concrete based approaches, while maintaining competitive costs and speeds, even at prototype-level quantities of parts. In Chapter 6, we use these building block systems to build a 2 m x 2 m x 2 m prototyping house element as a full-scale (though small) demonstrator, and use the same material system for reconfigurable furniture elements and two miniature cars.

Together, these results show that structural performance, error-correcting assembly, material-agnostic manufacturing, and electronic function can each be embedded into a single family of discrete building blocks. With this integration, the assembly robot can then be simple, low-cost, and even self-assembled, enabling parallelized, distributed assembly. This flips the conventional trajectory of large-scale fabrication, in which capability is concentrated in increasingly large and expensive machines, and instead distributes that capability into the matter being assembled. We have demonstrated robots can build not only functional structures, but also other robots, all using a single family of material systems. Using this system, we might be able to some day assemble sustainable buildings and infrastructure with integrated health monitoring, cars or drones where the wiring harness is directly integrated into the mechanical structure, or even in-space infrastructure, such as power utilities on the moon.

The underlying idea behind this is that by making physical material systems that behave like computational ones, we may eventually be able to unlock greater performance, design freedom, and production in the built world. This concept has been explored at the Center for Bits and Atoms for a long time, which really starts with Kenny Cheung's thesis, which showed that we could build precise, high-performance structures from discrete components. Revisiting the analogy between computation and physical materials, that work is like creating the structural, physical equivalent of a bit: a dependable primitive that we can build much larger systems on top of. Ben Jenett's thesis extended on that concept to show variable mechanical properties through non-load bearing assembly with simpler assemblers, which is somewhat akin to demonstrating an assembler, that is, a simple instruction set translating directly into a physical assembly with programmable material properties. Each step in this research lineage moves toward higher abstraction. What neither step addressed, however, was the integration of these dimensions simultaneously, which is what's needed to advance to practical application, especially in a construction context. This thesis takes the step of unifying structural performance, error-correcting assembly, material-agnostic manufacturability, and electronic function within a single family of blocks. This addition of these various functionalities into a single block system is not meant to only add new features, but rather the interaction of each of these properties with each other is what ultimately allows the system to scale, e.g. the structure carries the wiring, the wiring enables the robot, and the robot assembles the structure. I have extended this work so that the robot-material system now embodies sufficient logic to enable the cost-effective and fast assembly of load-bearing structures made from a wide variety of materials. In keeping with the computational analogy, this would be like developing a high-level language that works across materials, like Python, in which much of the design intelligence has been abstracted into the design of the system itself.

This work has so far been demonstrated at a lab scale and many critical functions remain either untested or undeveloped for real world deployment. Namely, our structural verification is first-order, and only considers a few basic loading cases. In a real structure, especially a building, much more complex loading would be expected, including some consideration of dynamic behavior, such as from seismic loads. Similarly, fire safety, thermal behavior, or humidity effects are not considered and are critical considerations for building-scale deployment. Certain features of the robotic system, such as how voxels or voxel parts are fed into the system, are not developed during this work, but would need to be automated for real deployment. The key features of this work have also been demonstrated by multiple instantiations of a family of discrete block types, as opposed to truly a single library of cross-compatible blocks. A future with force-aware bridges, or robots that merge into the built architecture, or arbitrarily robotically (and manually) reconfigurable housing would want a fully integrated version of this system, where within a single library of block types, we might have electronic, structural, and robotic blocks of varying dimensions, possibly all made from locally-sourced materials.

Future work looks toward addressing many of these gaps. The long-term performance of these structures in a relevant environment should be evaluated, as should new systems for insulation and for enclosing the structure. More complex and collaborative behaviors from the robots could be explored to further boost performance. Alternate, locally-produced material systems could be developed so that the voxel system might be arbitrarily deployable in any build site. Methods of post-tensioning, whether with cables or with structural skin systems, may provide a route toward highly material-efficient performance under tensile loads. Together, this additional work would enable the transition from these lab-scale demonstrations toward finished buildings or other infrastructure. Another parallel route of future work is to consider where the current building system, which is reconfigurable, can easily be assembled into almost arbitrary geometries, and does not have tooling costs, could be deployed, such as in temporary structures, forming, shoring, and other such applications.

As the technology continues to mature, it will also run into a regulatory environment that will need to be navigated, as well as other standards-based barriers to entry. Future work, in the lab and beyond, will need to look toward how to meet these regulations.

More broadly, this work points toward a construction paradigm in which the building block, rather than the building machine, carries the logic of what is made. This provides a path toward distributed, error-correcting, and self-scaling systems that build functional structures with less material, less infrastructure, and greater freedom to adapt, reuse, and rebuild over time. The goal of all of this work is to help realize a future where we can build more complex and performant structures in a sustainable way.

Bibliography

[1] Arduino Playground - CapacitiveSensor.

[2] LEGO® MINDSTORMS® | Invent a Robot | Official LEGO® Shop US.

[3] Sustainable Materials: With Both Eyes Open.

[4] I2C-bus specification and user manual.

[5] First 100% bio-based 3D-printed home unveiled at the University of Maine - UMaine News - University of Maine, November 2022.

[6] 3D Printed Rocket Launched Using Innovative NASA Alloy - NASA, May 2023. Section: Marshall Space Flight Center.

[7] Massachusetts Electricity Profile 2023, 2023.

[8] Branch Technology, 2024.

[9] House of Design, 2024.

[10] MX3D | Bridge, 2024.

[11] SAM, 2024.

[12] 3d Printer House | Crane WASP, 2025.

[13] Hadrian X® | Outdoor Construction & Bricklaying Robot from FBR, 2025.

[14] MaxiPrinter | Concrete 3D Printer | Constructions-3D, 2025.

[15] MetalsDepot® - Buy Steel Sheet Online - Any Quantity, Any Size!, 2025.

[16] Abdalla, H., Fattah, K. P., Abdallah, M., and Tamimi, A. K. Environmental Footprint and Economics of a Full-Scale 3D-Printed House. *Sustainability 13*, 21 (January 2021), 11978. Number: 21 Publisher: Multidisciplinary Digital Publishing Institute.

[17] Abdel-Rahman, A., Cameron, C., Jenett, B., Smith, M., and Gershenfeld, N. Self-replicating hierarchical modular robotic swarms. *Communications Engineering 1*, 1 (November 2022), 35.

[18] Abdullah, M., Seidel, L., Wernicke, B., Gouasmi, M., Hackl, A., Kern, T., Lempert, C., Lempert, C., Bizer, D., Storch, W., Fang, C., and Baudisch, P. Popcore: Personal fabrication of 3d foamcore models for professional high-quality applications in design and architecture. In *Proceedings of the 9th ACM Symposium on Computational Fabrication* (New York, NY, USA, 2024), SCF '24, Association for Computing Machinery.

[19] Abdullah, M., Sommerfeld, R., Sievers, B., Geier, L., Noack, J., Ding, M., Thieme, C., Seidel, L., Fritzsche, L., Langenhan, E., Adameck, O., Dzingel, M., Kern, T., Taraz, M., Lempert, C., Katakura, S., Elhassany, H. M., Roumen, T., and Baudisch, P. Hingecore: Laser-cut foamcore for fast assembly. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2022), UIST '22, Association for Computing Machinery.

[20] Afari, E., and Khine, M. Robotics as an Educational Tool: Impact of Lego Mindstorms. *International Journal of Information and Education Technology 7* (January 2017), 437–442.

[21] Agrawal, H., Umapathi, U., Kovacs, R., Frohnhofen, J., Chen, H.-T., Mueller, S., and Baudisch, P. Protopiper: Physically Sketching Room-Sized Objects at Actual Scale. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (New York, NY, USA, November 2015), UIST '15, Association for Computing Machinery, pp. 427–436.

[22] Agustí-Juan, I., and Habert, G. Environmental design guidelines for digital fabrication. *Journal of Cleaner Production 142* (January 2017), 2780–2791.

[23] Alalawi, M., Pacik-Nelson, N., Zhu, J., Greenspan, B., Doan, A., Wong, B. M., Owen-Block, B., Mickens, S. K., Schoeman, W. J., Wessely, M., Danielescu, A., and Mueller, S. MechSense: A Design and Fabrication Pipeline for Integrating Rotary Encoders into 3D Printed Mechanisms. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, April 2023), CHI '23, Association for Computing Machinery, pp. 1–14.

[24] American Society of Civil Engineers, Ed. *Minimum design loads for buildings and other structures*, 2. print ed. No. 7-10 in ASCE standard ASCE/SEI. ASCE, American Society of Civil Engineers, Reston, Va, 2011.

[25] Andjela, L., Abdurahmanovich, V. M., Vladimirovna, S. N., Mikhailovna, G. I., Yurievich, D. D., and Alekseevna, M. Y. A review on vat photopolymerization 3d-printing processes for dental application. *Dental Materials 38*, 11 (2022), e284–e296.

[26] Apolinarska, A. A., Knauss, M., Gramazio, F., and Kohler, M. The sequential roof. In *Advancing Wood Architecture*, 1st edition ed. Routledge, 2016, p. 14. eBook.

[27] Apolinarska, A. A., Pacher, M., Li, H., Cote, N., Pastrana, R., Gramazio, F., and Kohler, M. Robotic assembly of timber joints using reinforcement learning. *Automation in Construction 125* (2021), 103569.

[28] Augugliaro, F., Lupashin, S., Hamer, M., Male, C., Hehn, M., Mueller, M. W., Willmann, J. S., Gramazio, F., Kohler, M., and D'Andrea, R. The Flight Assembled Architecture installation: Cooperative construction with flying machines. *IEEE Control Systems Magazine 34*, 4 (August 2014), 46–64. Conference Name: IEEE Control Systems Magazine.

[29] Bachmann, M., Marxen, A., Schomäcker, R., and Bardow, A. High performance, but low cost and environmental impact? Integrated techno-economic and life cycle assessment of polyoxazolidinone as a novel high-performance polymer. *Green Chemistry 24*, 23 (2022), 9143–9156. Publisher: Royal Society of Chemistry.

[30] Batikha, M., Jotangia, R., Baaj, M. Y., and Mousleh, I. 3D concrete printing for sustainable and economical construction: A comparative study. *Automation in Construction 134* (February 2022), 104087.

[31] Baudisch, P., Silber, A., Kommana, Y., Gruner, M., Wall, L., Reuss, K., Heilman, L., Kovacs, R., Rechlitz, D., and Roumen, T. Kyub: A 3D Editor for Modeling Sturdy Laser-Cut Objects. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow Scotland Uk, May 2019), ACM, pp. 1–12.

[32] Bdeir, A. Electronics as material: littleBits. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction* (New York, NY, USA, February 2009), TEI '09, Association for Computing Machinery, pp. 397–400.

[33] Belke, C. H., and Paik, J. Mori: A Modular Origami Robot. *IEEE/ASME Transactions on Mechatronics 22*, 5 (October 2017), 2153–2164. Conference Name: IEEE/ASME Transactions on Mechatronics.

[34] Bello, N. D., and Memari, A. M. Comparative Review of the Technology and Case Studies of 3D Concrete Printing of Buildings by Several Companies. *Buildings 13*, 1 (Jan. 2023), 106. Publisher: Multidisciplinary Digital Publishing Institute.

[35] Bhundiya, H. G., and Cordero, Z. C. Bend-forming: A cnc deformation process for fabricating 3d wireframe structures. *Additive Manufacturing Letters 6* (2023), 100146.

[36] Bhundiya, H. G., and Cordero, Z. C. Bend-forming: A cnc deformation process for fabricating 3d wireframe structures. *Additive Manufacturing Letters 6* (2023), 100146.

[37] Bock, T. The future of construction automation: Technological disruption and the upcoming ubiquity of robotics. *Automation in Construction 59* (2015), 113–121.

[38] Bodea, S., Mindermann, P., Gresser, G. T., and Menges, A. Additive Manufacturing of Large Coreless Filament Wound Composite Elements for Building Construction. *3D Printing and Additive Manufacturing 9*, 3 (June 2022), 145–160. Publisher: Mary Ann Liebert, Inc., publishers.

[39] Buechley, L., Eisenberg, M., Catchen, J., and Crockett, A. The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, April 2008), CHI '08, Association for Computing Machinery, pp. 423–432.

[40] Built Robotics, Inc. Built robotics — robots that build the world. `https://www.builtrobotics.com/`, 2025. Accessed: 2025-12-25.

[41] By. Iowa Demolishes Its First 3D Printed House, November 2023.

[42] Cameron, C. G., Fredin, Z., and Gershenfeld, N. Discrete Assembly of Unmanned Aerial Systems. In *2022 International Conference on Unmanned Aircraft Systems (ICUAS)* (Dubrovnik, Croatia, June 2022), IEEE, pp. 339–344.

[43] Canvas. Canvas 1200CX: The World's First Compact Drywall Robot. `https://canvas.build/products/canvas1200cx/`, 2025. Accessed: 2025-12-25.

[44] Carney, M. E. *Discrete Cellular Lattice Assembly*. PhD, Massachusetts Institute of Technology, Cambridge MA, Sept. 2015.

[45] Chen, X. A., Tao, Y., Wang, G., Kang, R., Grossman, T., Coros, S., and Hudson, S. E. Forte: User-Driven Generative Design. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC Canada, April 2018), ACM, pp. 1–12.

[46] Cheung, K. C., and Gershenfeld, N. Reversibly Assembled Cellular Composite Materials. *Science 341*, 6151 (September 2013), 1219–1221.

[47] Chourasia, A., Singhal, S., and null Manivannan. Prefabricated volumetric modular construction: A review on current systems, challenges, and future prospects. *Practice Periodical on Structural Design and Construction 28*, 1 (2023), 03122009.

[48] ClarkDietrich. 600S162-54-P (50ksi, CP60, Punched), 2025.

[49] COBOD. EU COBOD BOD2 Tech Specs, 2022.

[50] Costa, A., Abdel-Rahman, A., Jenett, B., Gershenfeld, N., Kostitsyna, I., and Cheung, K. Algorithmic Approaches to Reconfigurable Assembly Systems. In *2019 IEEE Aerospace Conference* (Big Sky, MT, USA, March 2019), IEEE, pp. 1–8.

[51] Cramer, N. B., Cellucci, D. W., Formoso, O. B., Gregg, C. E., Jenett, B. E., Kim, J. H., Lendraitis, M., Swei, S. S., Trinh, G. T., Trinh, K. V., and Cheung, K. C. Elastic shape morphing of ultralight structures by programmable assembly. *Smart Materials and Structures 28*, 5 (apr 2019), 055006.

[52] Curth, A., Pearl, N., Castro-Salazar, A., Mueller, C., and Sass, L. 3D printing earth: Local, circular material processing, fabrication methods, and Life Cycle Assessment. *Construction and Building Materials 421* (March 2024), 135714.

[53] Do, S. serenado/proofOfDance, September 2021. original-date: 2019-04-08T20:44:55Z.

[54] Dong, Y. H., and Ng, S. T. A modeling framework to evaluate sustainability of building construction based on LCSA. *The International Journal of Life Cycle Assessment 21*, 4 (April 2016), 555–568.

[55] Dorsey, K. L., Roberts, S. F., Forman, J., and Ishii, H. Analysis of defextiles: a 3d printed textile towards garments and accessories. *Journal of Micromechanics and Microengineering 32*, 3 (feb 2022), 034005.

[56] Drieth, B. Hannah begins construction on "first multi-storey 3D-printed house in the US", October 2022. Section: Architecture.

[57] Enemuoh, E. U., Menta, V. G., Abutunis, A., O'Brien, S., Kaya, L. I., and Rapinac, J. Energy and Eco-Impact Evaluation of Fused Deposition Modeling and Injection Molding of Polylactic Acid. *Sustainability 13*, 4 (January 2021), 1875. Publisher: Multidisciplinary Digital Publishing Institute.

[58] EPA. 2019 Fiberglass Plant Carbon Intensities Fact Sheet .pdf, 2019.

[59] Eversmann, P., Ochs, J., Heise, J., Akbar, Z., and Böhm, S. Additive Timber Manufacturing: A Novel, Wood-Based Filament and Its Additive Robotic Fabrication Techniques for Large-Scale, Material-Efficient Construction. *3D Printing and Additive Manufacturing 9*, 3 (June 2022), 161–176. Publisher: Mary Ann Liebert, Inc., publishers.

[60] Florian, M.-C. ICON Reveals New 3D-Printed Residential Development in Wimberley, Texas | ArchDaily, 2024.

[61] Forman, J., Dogan, M. D., Forsythe, H., and Ishii, H. DefeXtiles: 3D Printing Quasi-Woven Fabric via Under-Extrusion. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, October 2020), UIST '20, Association for Computing Machinery, pp. 1222–1233.

[62] Forman, J., Kilic Afsar, O., Nicita, S., Lin, R. H.-J., Yang, L., Hofmann, M., Kothakonda, A., Gordon, Z., Honnet, C., Dorsey, K., Gershenfeld, N., and Ishii, H. Fiberobo: Fabricating 4d fiber interfaces by continuous drawing of temperature tunable liquid crystal elastomers. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2023), UIST '23, Association for Computing Machinery.

[63] Formlabs. How to create models larger than your 3d printer's build volume, 2023. Accessed: 2025-07-17.

[64] Fredin, Z., Zemanek, J., Blackburn, C., Strand, E., Abdel-Rahman, A., Rowles, P., and Gershenfeld, N. Discrete Integrated Circuit Electronics (DICE). In *2020 IEEE High Performance Extreme Computing Conference (HPEC)* (Waltham, MA, USA, September 2020), IEEE, pp. 1–8.

[65] Garcia, J., Yannuzzi, M., Kramer, P., Rieck, C., Fekete, S. P., and Becker, A. T. Reconfiguration of a 2D Structure Using Spatio-Temporal Planning and Load Transferring. In *2024 IEEE International Conference on Robotics and Automation (ICRA)* (May 2024), pp. 8735–8741.

[66] Gebler, M., Schoot Uiterkamp, A. J. M., and Visser, C. A global sustainability perspective on 3D printing technologies. *Energy Policy 74* (November 2014), 158–167.

[67] Georgiou, D., Okegbu, D., Yang, Z., Wang, T., Snowdon, M. R., Mohanty, A., Gershenfeld, N., Yan, W., and Athanasiou, C. E. Eco-voxels: Building blocks for sustainable, load-bearing structures. *Matter 8*, 7 (July 2025), 102106.

[68] Gharbia, M., Chang-Richards, A., Lu, Y., Zhong, R. Y., and Li, H. Robotic technologies for on-site building construction: A systematic review. *Journal of Building Engineering 32* (2020), 101584.

[69] Goessens, S., Mueller, C., and Latteur, P. Feasibility study for drone-based masonry construction of real-scale structures. *Automation in Construction 94* (October 2018), 458–480.

[70] Gong, J., Seow, O., Honnet, C., Forman, J., and Mueller, S. MetaSense: Integrating Sensing Capabilities into Mechanical Metamaterial. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, October 2021), UIST '21, Association for Computing Machinery, pp. 1063–1073.

[71] Gregg, C. E., Catanoso, D., Formoso, O. I. B., Kostitsyna, I., Ochalek, M. E., Olatunde, T. J., Park, I. W., Sebastianelli, F. M., Taylor, E. M., Trinh, G. T., and Cheung, K. C. Ultralight, strong, and self-reprogrammable mechanical metamaterials. *Science Robotics 9*, 86 (January 2024), eadi2746. Publisher: American Association for the Advancement of Science.

[72] Gregg, C. E., Kim, J. H., and Cheung, K. C. Ultra-Light and Scalable Composite Lattice Materials. *Advanced Engineering Materials 20*, 9 (2018), 1800213. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/adem.201800213.

[73] Hamann, H. *Swarm Robotics: A Formal Approach*, 1 ed. Springer, Cham, 2018.

[74] Hanton, O., Wessely, M., Mueller, S., Fraser, M., and Roudaut, A. ProtoSpray: Combining 3D Printing and Spraying to Create Interactive Displays with Arbitrary Shapes. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, April 2020), CHI '20, Association for Computing Machinery, pp. 1–13.

[75] Hauser, S., Mutlu, M., Léziart, P.-A., Khodr, H., Bernardino, A., and Ijspeert, A. Roombots extended: Challenges in the next generation of self-reconfigurable modular robots and their application in adaptive and assistive furniture. *Robotics and Autonomous Systems 127* (2020), 103467.

[76] Helm, V., Willmann, J., Gramazio, F., and Kohler, M. In-Situ Robotic Fabrication: Advanced Digital Manufacturing Beyond the Laboratory. In *Gearing Up and Accelerating Cross-fertilization between Academic and Industrial Robotics Research in Europe:* (Cham, 2014), F. Röhrbein, G. Veiga, and C. Natale, Eds., Springer Tracts in Advanced Robotics, Springer International Publishing, pp. 63–83.

[77] Hong, F., Myant, C., and Boyle, D. E. Thermoformed Circuit Boards: Fabrication of highly conductive freeform 3D printed circuit boards with heat bending. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, May 2021), CHI '21, Association for Computing Machinery, pp. 1–10.

[78] Hsu, A., Wong-Foy, A., McCoy, B., Cowan, C., Marlow, J., Chavez, B., Kobayashi, T., Shockey, D., and Pelrine, R. Application of micro-robots for building carbon fiber trusses. In *2016 International Conference on Manipulation, Automation and Robotics at Small Scales (MARSS)* (July 2016), pp. 1–6.

[79] Huang, M. Building a DIY Dance Dance Revolution, April 2019.

[80] Huang, Y., Carstensen, J. V., and Mueller, C. T. 3D truss topology optimization for automated robotic spatial extrusion. In *Proceedings of the IASS Symposium* (Boston, USA, July 2018), International Association for Shell and Spatial Structures.

[81] Huang, Y., Garrett, C., and Mueller, C. Constructability-driven design of frame structures with state-space search methods. *Automation in Construction 167* (2024), 105711.

[82] ICC. *2018 INTERNATIONAL BUILDING CODE (IBC)*. ICC, 2018.

[83] Institute, P. C. *PCI Design Handbook Precast and Prestressed Concrete*, 5th ed. Precast/Prestressed Concrete Institute, 1999.

[84] Institute for Advanced Architecture of Catalonia (IAAC). 3d printed bridge — iaac, 2017. accessed December 8, 2025.

[85] Insulating Concrete Forms Manufacturers Association. Insulating concrete forms manufacturers association. `https://icf-ma.org/`, 2025. Accessed: 2025-12-20.

[86] ISO. Environmental management – life cycle assessment – principles and framework. Tech. Rep. ISO 14040:2021, International Organization for Standardization, Geneva, Switzerland, 2021.

[87] ISO. Environmental management – life cycle assessment – requirements and guidelines. Tech. Rep. ISO 14044:2021, International Organization for Standardization, Geneva, Switzerland, 2021.

[88] JarrodB1. Inexpensive Hard DDR/Stepmania Dance Pad Controller.

[89] Jenett, B., Abdel-Rahman, A., Cheung, K., and Gershenfeld, N. Material–Robot System for Assembly of Discrete Cellular Structures. *IEEE Robotics and Automation Letters 4*, 4 (October 2019), 4019–4026.

[90] Jenett, B., Calisch, S., Cellucci, D., Cramer, N., Gershenfeld, N., Swei, S., and Cheung, K. C. Digital Morphing Wing: Active Wing Shaping Concept Using Composite Lattice-Based Cellular Structures. *Soft Robotics 4*, 1 (March 2017), 33–48. Publisher: Mary Ann Liebert, Inc., publishers.

[91] Jenett, B., Cameron, C., Tourlomousis, F., Rubio, A. P., Ochalek, M., and Gershenfeld, N. Discretely assembled mechanical metamaterials. *Science Advances 6*, 47 (November 2020).

[92] Jenett, B., Cellucci, D., Cheung, K., and Gregg, C. Meso-scale digital materials: Modular, reconfigurable, lattice-based structures. In *Proceedings of the 2016 Manufacturing Science and Engineering Conference (MSEC2016)* (Virginia Tech, Blacksburg, USA, 2016), ASME, pp. —. Paper MSEC2016-8767. Accessed at `https://cba.mit.edu/docs/papers/16.07.msec.bridge.pdf`.

[93] Jenett, B., and Cheung, K. BILL-E: Robotic Platform for Locomotion and Manipulation of Lightweight Space Structures. In *25th AIAA/AHS Adaptive Structures Conference* (Grapevine, Texas, January 2017), American Institute of Aeronautics and Astronautics.

[94] Jenett, B., Cheung, K., Gregg, C., and Cellucci, D. W. Design of multifunctional hierarchical space structures. In *2017 IEEE Aerospace Conference* (Big Sky, MT, USA, 2017), IEEE. NASA Document ID 20180007137. Accessed from `https://ntrs.nasa.gov/api/citations/20180007137/downloads/20180007137.pdf`.

[95] Jenett, B., Gershenfeld, N., and Guerrier, P. Building Block-Based Assembly of Scalable Metallic Lattices. In *Volume 4: Processes* (College Station, Texas, USA, June 2018), American Society of Mechanical Engineers, p. V004T03A053.

[96] Jenett, B. E. *Discrete mechanical metamaterials.* Thesis, Massachusetts Institute of Technology, 2020. Accepted: 2021-05-14T16:29:23Z.

[97] Jipa, A., and Dillenburger, B. 3D Printed Formwork for Concrete: State-of-the-Art, Opportunities, Challenges, and Applications. *3D Printing and Additive Manufacturing 9*, 2 (April 2022), 84–107. Publisher: Mary Ann Liebert, Inc., publishers.

[98] Jones, M. D., Anderson, Z., Walker, C., and Seppi, K. PHUI-kit: Interface Layout and Fabrication on Curved 3D Printed Objects. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC Canada, April 2018), ACM, pp. 1–11.

[99] Joshi, S. C., and Sheikh, A. A. 3D printing in aerospace and its long-term sustainability. *Virtual and Physical Prototyping 10*, 4 (October 2015), 175–185. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/17452759.2015.1111519.

[100] Kawajiri, K., and Sakamoto, K. Environmental impact of carbon fibers fabricated by an innovative manufacturing process on life cycle greenhouse gas emissions. *Sustainable Materials and Technologies 31* (April 2022), e00365.

[101] Keating, S., Spielberg, N. A., Klein, J., and Oxman, N. Digital Construction Platform: A Compound Arm Approach. *Keating* (2014). Accepted: 2014-11-14T16:50:10Z ISBN: 9783319046624 Publisher: Springer-Verlag.

[102] Kirby, B. T., Ashley-Rollman, M., and Goldstein, S. C. Blinky blocks: a physical ensemble programming platform. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems* (Vancouver BC Canada, May 2011), ACM, pp. 1111–1116.

[103] Kovacs, R., Ion, A., Lopes, P., Oesterreich, T., Filter, J., Otto, P., Arndt, T., Ring, N., Witte, M., Synytsia, A., and Baudisch, P. TrussFormer: 3D Printing Large Kinetic Structures. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, October 2018), UIST '18, Association for Computing Machinery, pp. 113–125.

[104] Kovacs, R., Rambold, L., Fritzsche, L., Meier, D., Shigeyama, J., Katakura, S., Zhang, R., and Baudisch, P. Trusscillator: a System for Fabricating Human-Scale Human-Powered Oscillating Devices. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, October 2021), UIST '21, Association for Computing Machinery, pp. 1074–1088.

[105] Kovacs, R., Seufert, A., Wall, L., Chen, H.-T., Meinel, F., Müller, W., You, S., Brehm, M., Striebel, J., Kommana, Y., Popiak, A., Bläsius, T., and Baudisch, P. TrussFab: Fabricating Sturdy Large-Scale Structures on Desktop 3D Printers. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, May 2017), CHI '17, Association for Computing Machinery, pp. 2606–2616.

[106] Kurokawa, H., Tomita, K., Kamimura, A., Kokaji, S., Hasuo, T., and Murata, S. Distributed Self-Reconfiguration of M-TRAN III Modular Robotic System. *The International Journal of Robotics Research 27*, 3-4 (March 2008), 373–386. Publisher: SAGE Publications Ltd STM.

[107] Kyaw, A. H., Jeon, S. H., Smith, M., and Gershenfeld, N. Making physical objects with generative AI and robotic assembly: Considering fabrication constraints, sustainability, time, functionality, and accessibility, 2025. Preprint, presented at the CHI 2025 Workshop on Generative AI and Human–Computer Interaction.

[108] Lambrichts, M., Ramakers, R., Hodges, S., Coppers, S., and Devine, J. A Survey and Taxonomy of Electronics Toolkits for Interactive and Ubiquitous Device Prototyping. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 5*, 2 (June 2021), 70:1–70:24.

[109] Lauer, A. P. R., Benner, E., Stark, T., Klassen, S., Abolhasani, S., Schroth, L., Gienger, A., Wagner, H. J., Schwieger, V., Menges, A., and

Sawodny, O. Automated on-site assembly of timber buildings on the example of a biomimetic shell. *Automation in Construction 156* (2023), 105118.

[110] Le Gargasson, P., Pelcat, M., Beuve, N., Pressigout, M., Morin, L., and Menard, D. PCBnCO: A Carbon Intensity Model of FR-4 Printed Circuit Boards Based on Company Data. In *2025 IEEE Conference on Technologies for Sustainability (SusTech)* (Los Angeles, CA, United States, Apr. 2025), IEEE.

[111] Leamon, S. Computational design of architected lattices for construction applications. Master's thesis, MASSACHUSETTS INSTITUTE OF TECHNOLOGY, Cambridge, MA, USA, June 2025. Any additional info (optional).

[112] Leder, S., Kim, H., Oguz, O. S., Kubail Kalousdian, N., Hartmann, V. N., Menges, A., Toussaint, M., and Sitti, M. Leveraging Building Material as Part of the In-Plane Robotic Kinematic System for Collective Construction. *Advanced Science 9*, 24 (2022), 2201524. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/advs.202201524.

[113] Leung, P. Y., Apolinarska, A. A., Tanadini, D., Gramazio, F., and Kohler, M. Automatic assembly of jointed timber structure using distributed robotic clamps. In *PROJECTIONS: Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design* (2021), vol. 1, Association for Computer Aided Architectural Design Research in Asia, pp. 583–592.

[114] Li, M., Li, D., Zhang, J., Cheng, J. C., and Gan, V. J. 3d printing in modular construction: Opportunities and challenges. *International conference on construction engineering and project management* (12 2020), 75–84.

[115] Liu, J., Nguyen-Van, V., Panda, B., Fox, K., du Plessis, A., and Tran, P. Additive Manufacturing of Sustainable Construction Materials and Form-finding Structures: A Review on Recent Progresses. *3D Printing and Additive Manufacturing 9*, 1 (February 2022), 12–34. Publisher: Mary Ann Liebert, Inc., publishers.

[116] Luban3D. Luban 3d, 2025. Accessed: 2025-07-17.

[117] Ma, Z., Walzer, A., Schumacher, C., Rust, R., Gramazio, F., Kohler, M., and Bächer, M. Designing robotically-constructed metal frame structures. *Computer Graphics Forum 39*, 2 (2020), 411–422.

[118] Marchment, T., and Sanjayan, J. Mesh reinforcing method for 3D Concrete Printing. *Automation in Construction 109* (January 2020), 102992.

[119] Meisel, N. A., Watson, N., Bilén, S. G., Duarte, J. P., and Nazarian, S. Design and System Considerations for Construction-Scale Concrete Additive Manufacturing in Remote Environments via Robotic Arm Deposition. *3D Printing and Additive Manufacturing 9*, 1 (February 2022), 35–45. Publisher: Mary Ann Liebert, Inc., publishers.

[120] Melenbrink, N., Kassabian, P., Menges, A., and Werfel, J. Towards Force-aware Robot Collectives for On-site Construction.

[121] Melenbrink, N., Wang, A., and Werfel, J. An Autonomous Vault-Building Robot System for Creating Spanning Structures. In *2021 IEEE International Conference on Robotics and Automation (ICRA)* (May 2021), pp. 7066–7072. ISSN: 2577-087X.

[122] Melenbrink, N., Werfel, J., and Menges, A. On-site autonomous construction robots: Towards unsupervised building. *Automation in Construction 119* (2020), 103312.

[123] Melenbrink, N., Werfel, J., and Menges, A. On-site autonomous construction robots: Towards unsupervised building. *Automation in Construction 119* (November 2020), 103312.

[124] Menges, A., Sheil, B., Glynn, R., and Skavara, M. *Fabricate 2017*. UCL Press, April 2017.

[125] Mirjan, A., Augugliaro, F., D'Andrea, R., Gramazio, F., and Kohler, M. Building a bridge with flying robots. In *Robotic Fabrication in Architecture, Art and Design 2016*, D. Reinhardt, R. Saunders, and J. Burry, Eds. Springer International Publishing, 2016, pp. 34–47.

[126] Mohammad, M., Masad, E., and Al-Ghamdi, S. G. 3D Concrete Printing Sustainability: A Comparative Life Cycle Assessment of Four Construction Method Scenarios. *Buildings 10*, 12 (December 2020), 245. Publisher: Multidisciplinary Digital Publishing Institute.

[127] Mugahed Amran, Y., El-Zeadani, M., Huei Lee, Y., Yong Lee, Y., Murali, G., and Feduik, R. Design innovation, efficiency and applications of structural insulated panels: A review. *Structures 27* (2020), 1358–1379.

[128] Nerella, V. N., Krause, M., and Mechtcherine, V. Direct printing test for buildability of 3D-printable concrete considering economic viability. *Automation in Construction 109* (January 2020), 102986.

[129] Neubert, J., and Lipson, H. Soldercubes: a self-soldering self-reconfiguring modular robot system. *Autonomous Robots 40*, 1 (January 2016), 139–158.

[130] Nicholson, S. R., Rorrer, N. A., Carpenter, A. C., and Beckham, G. T. Manufacturing energy and greenhouse gas emissions associated with plastics consumption. *Joule 5*, 3 (March 2021), 673–686.

[131] Nigl, F., Shuguang Li, Blum, J. E., and Lipson, H. Structure-Reconfiguring Robots: Autonomous Truss Reconfiguration and Manipulation. *IEEE Robotics & Automation Magazine 20*, 3 (September 2013), 60–71.

[132] Nisser, M., Cheng, L., Makaram, Y., Suzuki, R., and Mueller, S. ElectroVoxel: Electromagnetically Actuated Pivoting for Scalable Modular Self-Reconfigurable Robots. In *2022 International Conference on Robotics and Automation (ICRA)* (May 2022), pp. 4254–4260.

[133] Nisser, M., Liao, C. C., Chai, Y., Adhikari, A., Hodges, S., and Mueller, S. LaserFactory: A Laser Cutter-based Electromechanical Assembly and Fabrication Platform to Make Functional Devices & Robots. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, May 2021), CHI '21, Association for Computing Machinery, pp. 1–15.

[134] on Precast Sandwich Wall Panels, P. C. Sandwich Wall Panels Guide. Tech. rep., Precast/Prestressed Concrete Institute, 1997.

[135] Opgenorth, N., Locatelli, D. N., Leder, S., Wagner, H. J., and Menges, A. Multi-scalar robotic fabrication system for on-site press gluing in multi-storey timber buildings. *Automation in Construction 168* (2024), 105774.

[136] Parascho, S., Gandia, A., Mirjan, A., Gramazio, F., and Kohler, M. Cooperative fabrication of spatial metal structures. In *Fabricate 2017: Rethinking Design and Construction* (2017), UCL Press, pp. 24–29.

[137] Park, I.-W., Catanoso, D., Formoso, O., Gregg, C., Ochalek, M., Olatunde, T., Sebastianelli, F., Spino, P., Taylor, E., Trinh, G., and Cheung, K. SOLL-E: A Module Transport and Placement Robot for Autonomous Assembly of Discrete Lattice Structures. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Detroit, MI, USA, October 2023), IEEE, pp. 10736–10741.

[138] Parra Rubio, A., Fan, D., Jenett, B., Del Águila Ferrandis, J., Tourlomousis, F., Abdel-Rahman, A., Preiss, D., Zemánek, J., Triantafyllou, M., and Gershenfeld, N. Modular Morphing Lattices for Large-Scale Underwater Continuum Robotic Structures. *Soft Robotics 10*, 4 (August 2023), 724–736.

[139] Parra Rubio, A., Mundilova, K., Preiss, D., Demaine, E. D., and Gershenfeld, N. Kirigami corrugations: Strong, modular, and programmable plate lattices. In *Volume 8: 47th Mechanisms and Robotics Conference (MR)* (08 2023), International Design Engineering Technical Conferences and Computers and Information in Engineering Conference.

[140] PensaLabs. Pensa labs, 2025. Accessed: 2025-07-17.

[141] Perteneder, F., Probst, K., Leong, J., Gassler, S., Rendl, C., Parzer, P., Fluch, K., Gahleitner, S., Follmer, S., Koike, H., and Haller, M. Foxels: Build Your Own Smart Furniture. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction* (Sydney NSW Australia, February 2020), ACM, pp. 111–122.

[142] Petersen, K., Nagpal, R., and Werfel, J. TERMES: An Autonomous Robotic System for Three-Dimensional Collective Construction. *Robotics: Science and Systems* (2011).

[143] Petersen, K., Nagpal, R., and Werfel, J. K. Termes: An autonomous robotic system for three-dimensional collective construction. In *Robotics: Science and Systems*

VII, H. Durrant-Whyte, N. Roy, and P. Abbeel, Eds. MIT Press, Cambridge, MA, 2011.

[144] Petersen, K. H., Napp, N., Stuart-Smith, R., Rus, D., and Kovac, M. A review of collective robotic construction. *Science Robotics 4*, 28 (March 2019), eaau8479. Publisher: American Association for the Advancement of Science.

[145] Polymaker. Polymaker's "Liuyun Bridge" - Polymaker, 2021.

[146] Prado, M., Dörstelmann, M., Solly, J., Menges, A., and Knippers, J. Elytra filament pavilion: Robotic filament winding for structural composite building systems. In *Fabricate 2017: Rethinking Design and Construction* (2017), A. Menges, B. Sheil, R. Glynn, and M. Skavara, Eds., UCL Press, pp. 224–231.

[147] Price, L., Phylipsen, G., and Worrell, E. Energy use and carbon dioxide emissions in the steel sector in key developing countries. Tech. Rep. LBNL–46987, 783473, OSTI, Apr. 2001.

[148] Pun, A., Deng, K., Liu, R., Ramanan, D., Liu, C., and Zhu, J.-Y. Generating physically stable and buildable brick structures from text, 2025.

[149] Read, J. R., Mcelroy, L., Bolsee, Q., Smith, B., and Gershenfeld, N. Modular-Things: Plug-and-Play with Virtualized Hardware. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, April 2023), CHI EA '23, Association for Computing Machinery, pp. 1–6.

[150] Roux, C., Kuzmenko, K., Roussel, N., Mesnil, R., and Feraille, A. Life cycle assessment of a concrete 3D printing process. *The International Journal of Life Cycle Assessment 28*, 1 (January 2023), 1–15.

[151] RSMeans Gordian. RSMeans Data Online, 2018.

[152] Rubio, A. P., Mundilova, K., Preiss, D., Demaine, E., and Gershenfeld, N. Kirigami corrugations: Strong, modular and programmable plate lattices. In *Proceedings of the ASME 2023 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC-CIE2023)* (2023), ASME.

[153] S/A, G. EPD MBQ Gerdau Cambridge, 2025.

[154] Salamanca, L., Apolinarska, A. A., Pérez-Cruz, F., and Kohler, M. Augmented intelligence for architectural design with conditional autoencoders: Semiramis case study. In *Towards Radical Regeneration* (Cham, 2023), C. Gengnagel, O. Baverel, G. Betti, M. Popescu, M. R. Thomsen, and J. Wurm, Eds., Springer International Publishing, pp. 108–121.

[155] Sass, L., and Botha, M. The instant house: A model of design production with digital fabrication. *International Journal of Architectural Computing 4*, 4 (2006), 109–123.

[156] Schaedler, T. A., and Carter, W. B. Architected Cellular Materials. *Annual Review of Materials Research 46*, 1 (2016), 187–210. _eprint: https://doi.org/10.1146/annurev-matsci-070115-031624.

[157] Schaedler, T. A., Jacobsen, A. J., Torrents, A., Sorensen, A. E., Lian, J., Greer, J. R., Valdevit, L., and Carter, W. B. Ultralight Metallic Microlattices. *Science 334*, 6058 (November 2011), 962–965.

[158] Shannon, C. Communication in the presence of noise. *Proceedings of the IRE 37*, 1 (1949), 10–21.

[159] Siriwardena, L., Stark, T., Lut, S., Wagner, H. J., Maierhofer, M., Bechert, S., Knippers, J., and Menges, A. Joint effort - a material-robot system for fibrous joints of lightweight timber construction. In *Proceedings of the 9th ACM Symposium on Computational Fabrication* (New York, NY, USA, 2024), SCF '24, Association for Computing Machinery.

[160] Skuric, A., Bank, H. S., Unger, R., Williams, O., and González-Reyes, D. Simplefoc: A field oriented control (foc) library for controlling brushless direct current (bldc) and stepper motors. *Journal of Open Source Software 7*, 74 (2022), 4232.

[161] Smith, M., Abdel-Rahman, A., and Gershenfeld, N. Self-Reconfigurable Robots for Collaborative Discrete Lattice Assembly. In *2024 IEEE International Conference on Robotics and Automation (ICRA)* (May 2024), pp. 3624–3631.

[162] Smith, M., Forman, J., Abdel-Rahman, A., Wang, S., and Gershenfeld, N. Voxel Invention Kit: Reconfigurable Building Blocks for Prototyping Interactive Electronic Structures. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, Apr. 2025), CHI '25, Association for Computing Machinery, pp. 1–15.

[163] Smith, M., Richard, P., Kyaw, A., and Gershenfeld, N. Hierarchical discrete lattice assembly: An approach for the digital fabrication of scalable macroscale structures. In *ACM Symposium on Computational Fabrication (SCF '25)* (New York, NY, USA, 2025), ACM, p. 15.

[164] Smith, M., Richard, P., Rubio, A. P., and Gershenfeld, N. Comparative evaluation of robotically assembled discrete lattice systems for sustainable construction. *Automation in Construction 187* (2026), 106952.

[165] Smith, M. M. *Recursive Robotic Assemblers.* Thesis, Massachusetts Institute of Technology, June 2023. Accepted: 2023-08-30T16:00:19Z.

[166] Sonebi, M., Ammar, Y., and Diederich, P. 15 - Sustainability of cement, concrete and cement replacement materials in construction. In *Sustainability of Construction Materials (Second Edition)*, J. M. Khatib, Ed., Woodhead Publishing Series in Civil and Structural Engineering. Woodhead Publishing, January 2016, pp. 371–396.

[167] Spröwitz, A., Moeckel, R., Vespignani, M., Bonardi, S., and Ijspeert, A. J. Roombots: A hardware perspective on 3D self-reconfiguration and locomotion with a homogeneous modular robot. *Robotics and Autonomous Systems 62*, 7 (July 2014), 1016–1033.

[168] Swissler, P., and Rubenstein, M. FireAnt3D: a 3D self-climbing robot towards non-latticed robotic self-assembly. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Las Vegas, NV, USA, October 2020), IEEE, pp. 3340–3347.

[169] Tetmet SAS. Adaptive spatial lattice manufacturing (aslm). `https://www.tetmet.net/aslm`, 2025. Accessed: 2025-12-29.

[170] Tilt-Up Concrete Association. Construction basics. `https://tilt-up.org/construction/basics/`, n.d. Accessed: 2025-12-20.

[171] Vasilevitsky, T., and Zoran, A. Steel-Sense: Integrating Machine Elements with Sensors by Additive Manufacturing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, May 2016), CHI '16, Association for Computing Machinery, pp. 5731–5742.

[172] Wagner, H. J., Alvarez, M., Kyjanek, O., Bhiri, Z., Buck, M., and Menges, A. Flexible and transportable robotic timber construction platform – tim. *Automation in Construction 120* (2020), 103400.

[173] Waldman, B., Hyatt, A., Carlisle, S., Palmeri, J., and Simonen, K. 2023 Carbon Leadership Forum North American Material Baselines (Version 2). Tech. rep., Carbon Leadership Forum, University of Washington, Seattle, WA, 2023.

[174] Wang, S., Smith, M., and Gershenfeld, N. Voxelcopter: Modular Autonomous Aerial Systems. In *Proceedings of the 8th ACM Symposium on Computational Fabrication* (New York City NY USA, October 2023), ACM, pp. 1–2.

[175] Weng, Y., Li, M., Ruan, S., Wong, T. N., Tan, M. J., Ow Yeong, K. L., and Qian, S. Comparative economic, environmental and productivity assessment of a concrete bathroom unit fabricated through 3D printing and a precast approach. *Journal of Cleaner Production 261* (July 2020), 121245.

[176] Wessely, M., Sethapakdi, T., Castillo, C., Snowden, J. C., Hanton, O., Qamar, I. P. S., Fraser, M., Roudaut, A., and Mueller, S. Sprayable User Interfaces: Prototyping Large-Scale Interactive Surfaces with Sensors and Displays. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, April 2020), CHI '20, Association for Computing Machinery, pp. 1–12.

[177] Xu, Z., and Dollar, A. M. Chain-based lattice printing for efficient robotically-assembled structures. *Communications Engineering 3* (2024), 157.

[178] Yim, M., Duff, D., and Roufas, K. PolyBot: a modular reconfigurable robot. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)* (April 2000), vol. 1, pp. 514–520 vol.1. ISSN: 1050-4729.

[179] Yoon, Y., and Rus, D. Shady3D: A Robot that Climbs 3D Trusses. In *Proceedings 2007 IEEE International Conference on Robotics and Automation* (Rome, Italy, April 2007), IEEE, pp. 4071–4076. ISSN: 1050-4729.

[180] Yu, J., Sakhardande, P., Parmar, R., and Oh, H. Strawctures: A Modular Electronic Construction Kit for Human-Scale Interactive Structures. In *Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction* (New York, NY, USA, February 2022), TEI '22, Association for Computing Machinery, pp. 1–10.

[181] Zhang, N., and Sanjayan, J. Concrete 3D printing and digital fabrication technologies for bridge construction. *Automation in Construction 179* (November 2025), 106485.

[182] Zhang, X., Li, M., Lim, J. H., Weng, Y., Tay, Y. W. D., Pham, H., and Pham, Q.-C. Large-scale 3D printing by a team of mobile robots. *Automation in Construction 95* (November 2018), 98–106.

[183] Zheng, C., Han, B., Liu, X., Devendorf, L., Tan, H., and Yen, C. C. Crafting Interactive Circuits on Glazed Ceramic Ware. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, April 2023), CHI '23, Association for Computing Machinery, pp. 1–18.

[184] Zheng, X., Lee, H., Weisgraber, T. H., Shusteff, M., DeOtte, J., Duoss, E. B., Kuntz, J. D., Biener, M. M., Ge, Q., Jackson, J. A., Kucheyev, S. O., Fang, N. X., and Spadaccini, C. M. Ultralight, ultrastiff mechanical metamaterials. *Science 344*, 6190 (June 2014), 1373–1377. Publisher: American Association for the Advancement of Science.

[185] Zhu, J., Blumberg, L.-G., Zhu, Y., Nisser, M., Carlson, E. L., Wen, X., Shum, K., Quaye, J. A., and Mueller, S. CurveBoards: Integrating Breadboards into Physical Objects to Prototype Function in the Context of Form. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, April 2020), CHI '20, Association for Computing Machinery, pp. 1–13.

[186] Zhu, J., Zhu, Y., Cui, J., Cheng, L., Snowden, J., Chounlakone, M., Wessely, M., and Mueller, S. MorphSensor: A 3D Electronic Design Tool for Reforming Sensor Modules. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, October 2020), UIST '20, Association for Computing Machinery, pp. 541–553.

[187] Zuo, Z., De Corte, W., Huang, Y., Chen, X., Zhang, Y., Li, J., Zhang, L., Xiao, J., Yuan, Y., Zhang, K., Zhang, L., and Mechtcherine, V. Pro-

pelling the widespread adoption of large-scale 3D printing. *Nature Reviews Materials* (November 2023), 1–3. Publisher: Nature Publishing Group.