

## **Ultra-light, strong, and self-reprogrammable mechanical metamaterials**

This Author Accepted Manuscript (AAM) is the version of this paper published by AAAS, including changes resulting from peer review but prior to AAAS's copy editing and production). This is the author's version of the work. It is posted here per <https://sti.nasa.gov/disclaimers/>. The definitive version was published in Science Robotics, Vol 9, Issue 86, 10.1126/scirobotics.adi2746

Barrier free access to the Final Published Version of this paper is available here <https://www.science.org/stoken/author-tokens/ST-1698/full>, with NTRS serving as the corresponding author website for which AAAS provides this link. Starting in January 2025, <https://www.science.org/doi/10.1126/scirobotics.adi2746> will also provide barrier free access to the Final Published Version.

We thank AAAS for supporting our efforts to explore the unknown in air and space, innovate for the benefit of humanity, and inspire the world through discovery.

**Title:** Ultra-light, strong, and self-reprogrammable mechanical metamaterials

**Authors:**

Christine E. Gregg,<sup>1\*</sup> Damiana Catanoso,<sup>2</sup> Olivia Irene B. Formoso,<sup>1</sup> Irina Kostitsyna,<sup>2</sup> Megan E. Ochalek,<sup>1</sup> Taiwo J. Olatunde,<sup>2</sup> In Won Park,<sup>2</sup> Frank M. Sebastianelli,<sup>2</sup> Elizabeth M. Taylor,<sup>1</sup> Greenfield T. Trinh,<sup>1</sup> Kenneth C. Cheung<sup>1\*</sup>

**Affiliations:**

<sup>1</sup>NASA Ames Research Center, Moffett Field, CA, USA.

<sup>2</sup>KBR, Moffett Field, CA, USA

\*Corresponding author. Email: kenny@nasa.gov, christine.e.gregg@nasa.gov

**Abstract:**

Versatile programmable materials have long been envisioned that can reconfigure themselves to adapt to changing use cases in adaptive infrastructure, space exploration, disaster response, and more. We introduce a robotic structural system as an implementation of programmable matter, with mechanical performance and scale on par with conventional high-performance materials and truss systems. Fiber reinforced composite truss-like building-blocks form strong, stiff, and lightweight lattice structures as mechanical metamaterials. Two types of mobile robots operate over the exterior surface and through the interior of the system, performing transport, placement, and reversible fastening using the intrinsic lattice periodicity for indexing and metrology. Leveraging programmable matter algorithms to achieve scalability in size and complexity, this system design enables robust collective automated assembly and reconfiguration of large structures with simple robots. We describe the system design and experimental results from a 256-unit cell assembly demonstration and lattice mechanical testing, as well as demonstration of disassembly and reconfiguration. The assembled structural lattice material exhibits ultra-light mass density (0.0103 grams per cubic centimeter) with high strength and stiffness for its weight (11.17 kilopascals and 1.1129 megapascals, respectively), a material performance realm appropriate for applications like space structures. With simple robots and structure, high mass-specific structural performance, and competitive throughput, this system demonstrates potential for self-reconfiguring autonomous metamaterials for diverse applications.

**One-Sentence Summary:** Programmable matter can reconfigure and adapt autonomously, extending to high-performance mechanical materials at scale.

**Main Text:**

**INTRODUCTION**

Self-assembling and reconfigurable machines have long been proposed as material systems with the versatility and adaptability of general computers (1). Such systems promise the ability to adapt their properties and form as needed to meet changing needs, use-cases, or environments. Since they can be re-used and self-repaired, such systems could transform material life cycles for a more sustainable engineered world (2) and enable new engineering paradigms for zero-mass space exploration. Powerful advances in materials and methods demonstrate aspects of this vision as programmable materials in the form of phononic crystals (3), color-changing metamaterials (4), self-folding surfaces (5), and DNA self-assembly (6).

To realize macro-scale three-dimensional applications such as adaptive infrastructure, space-applications (7), disaster response, and more (8), programmable material systems are needed that can both reconfigure themselves at scale and meet high-performance mechanical requirements (in both tension and compression).

To progress towards this vision of macro-scale high-performance programmable materials, we present a system that combines desirable characteristics of collective robotic assembly, programmable matter theory and algorithms, and architected cellular solids to yield a self-reprogrammable truss system. The term ‘programmable matter’ is used in the literature to describe a broad range of systems with very different characteristics, from phononic crystals to reconfiguring bricks. Towards the spirit of general macro-scale three-dimensional reconfigurable materials, we distinguish programmable material examples that are fully three-dimensional, ‘self-programmable’, and ‘reprogrammable’. For material generality, we restrict consideration to systems that yield a three-dimensional material that has non-zero compression and tensile strength (placement or reconfiguration of unbonded bricks is not considered since it can’t be reconfigured to withstand tensile loading). We define self-programmable material systems as those that can reconfigure themselves using mechanisms, actuators, metrology, and reference frames intrinsic to the system. Systems that are not self-reprogrammable rely on external actuation, metrology, and/or reference frames (such as robotics or gantry systems that rely on global metrology and reference frame to reprogram elements). We term a system ‘reprogrammable’ if the system can reconfigure after initial manufacturing, rather than having a fixed behavior or configuration after initial programming (9–11). Systems that are both reconfigurable and rely only on intrinsic actuation and metrology (self-reprogrammable) have meaningful advantages in system scalability and error correction (1, 12) and enable efficient leverage of programmable matter theory and algorithms (13).

The fields of self-reconfigurable robots and collective robotic construction lead progress on fully synthetic three-dimensional self-reprogrammability (7, 14), but open challenges remain in realms of scalability and mechanical performance. Current self-reprogrammable material examples often rely on magnetic connections for alignment and holding forces (14–19), which have insufficient strength density to provide high-performance structural connections. For systems that rely on mechanical coupling with actuation or thermal bonding systems incorporated within each module, large system mass penalties govern the specific strength and stiffness of the resulting functional structure (14, 18, 20–23). Prior system designs for robot assembly and locomotion on truss structures (24–27) suggest a future goal of operational demonstration at scale. Reconfiguring truss robots, both self-reconfiguring (22) and manually reconfigured (28), have faced similar implementation challenges, likely due to the challenges of joint design and interface alignment.

Our self-reprogrammable system combines mechanical metamaterial building blocks and two types of robots (29, 30) to create a reconfigurable structural-robotic system (fig. S1). We separate the actuation and structural components to achieve system mass-efficiency on par with static structural materials (17, 21, 24–27). The use of relative robotic principles (31) ensures that this separation does not sacrifice the indexing and metrology associated with self-actuated lattice blocks. Each structural unit cell, termed ‘voxel’ as volumetric pixel, is a mechanical metamaterial

building-block that is mass-produced from high-performance fiber-reinforced composite materials. These building-blocks can be used to achieve a wide range of material properties by recombination of constituent material and geometry (32, 33). Simple robots use passive alignment features to locomote on the structure and index to each unit cell, achieving high precision locomotion within a reliable local reference system with low robotic sensing and control requirements. No vision or external metrology systems are utilized. Alignment features on the voxels and robots ensure that each aligns correctly to each other, providing robust reconfiguration and assembly. The functionality and design of the structure and robots are so closely intertwined that the robots retain little functionality without the structure, truly forming a joint robotic-structural system that pushes complexity of autonomous robotic structural assembly from hardware to software (planning and scheduling algorithms without complex perception needs).

## RESULTS

Movie 1 presents a system overview of the results in this paper. The following sections describe the system in detail.

### System Components

The cuboctahedron structural unit cell voxels (Fig. 1 (C)) were constructed from six injection molded chopped carbon fiber reinforced polymer (StattechNN-40CF) square faces (Fig. 1 (A) and fig S2) (corner to corner dimension of 304.8mm, yielding a voxel with a bounding box edge dimension of 304.8mm). The Material and Methods section and supplementary figures (fig. S3 and fig. S4) describe characterization of the constituent fiber reinforced injection-molded material (data in table S5 and table S6). These square faces were pre-assembled into voxels using conventional bolts and integrated alignment features (Fig. 1 (B)) located at the corners of the faces. Voxels connected face-to-face (Fig. 1 (D)) using four custom injection-molded genderless reversible fasteners (34) (Fig. 1 (F)) that were held captive in the voxel face (Fig. 1 (E)). O-rings between the fastener and voxel face housing ensured that the pre-installed fasteners were held captive in the open position during handling. Fasteners were actuated by the fastener robot end effectors, and when locked they supported a 797.3 N ( $\pm$  8.71 N) tension load (fig. S5 and table S1). They were designed to apply a preload to the intervoxel joint that increased the joint strength beyond the fastener load (like any preloaded bolted joint).

Two external transport robots (Fig. 2) collaborated to deliver voxels from a supply depot (voxel source) to the build front and place voxels into position. Each transport robot (30) operated as a bipedal inchworm with three main drive joints, two feet each capable of aligning to and gripping voxel faces, and two basal plane rotation stages (yaw drives) at the base of each foot for turning (Fig. 2 (A)). To align to voxels, the robot utilized alignment petal guides during locomotion and gripper claws during grasping. Each transport robot served one of two roles, designated as ‘cargo’ or ‘crane’ service. The cargo robot featured a third gripper (‘cargo gripper’) to hold a voxel while moving across the structure. To locomote, the transport robot gripped one foot to the structure while the other foot accessed nearby voxels faces on either the same level, one level down, or one level up, alternating foot gripper states at these configurations to inchworm freely over a stepped surface (Fig. 2 (D)).

The crane robot could unload a voxel from the cargo gripper (Fig. 2(B)) and place or remove a voxel at any available lattice position (Fig. 2(C)). It did this by using one of its gripper feet as an end effector to act like a mobile robot arm.

An internal fastening robot (29) (Fig. 3 and fig. S6) climbed through the structure and actuated captive fasteners between voxels (Fig. 3 (E)). In the contracted position (Fig. 3(B)), the robot occupied a square face of the cuboctahedral lattice. By extending (Fig. 3 (A) and Fig. 3 (C)) and gripping adjacent faces (Fig. 3 (D)), the robot moved between faces and traversed in three dimensions (Fig. 3 (G)). Dual purpose alignment features on the robot body (skis) and grippers (claws) (Fig. 3 (F)) ensured the robot always aligned the unfastened voxel and mated the aligned face via gripping of the newly placed voxel. When both faces were gripped by the claws, the unfastened voxel aligned for fastening to the existing voxel. While in the contracted position and gripping two voxel faces, eight ‘bolting’ modules actuated the fasteners at the interface between voxels to either connect or disconnect units for assembly or disassembly (for reconfiguration or repair). Further robot details are available in the Materials and Methods section.

The robots had multi-stage alignment guides that worked in combination with robot compliance to ensure proper and robust positioning of end effectors during operation. Some features assisted alignment during robot locomotion, and some assisted alignment during voxel grabbing. It was permissible for the placement/locomotion accuracy in the initial stages to be relatively imprecise, as the use of alignment guides in subsequent stages improved precision. This was because the system’s capture envelopes overlapped with decreasing size. As long as the initial alignment was inside of the initial envelope, it was guaranteed that subsequent alignment stages would further increase voxel positioning accuracy and precision.

When the external robot took a step or grabbed a voxel for placement, the alignment petals could accommodate 3.0 cm in-plane misalignment, and the external robot grippers could accommodate a 1.34 cm out-of-plane offset from the gripped face. During placement, the cargo transport robot was required to place the voxel within +/- 1.78 cm (in X, Y, and Z) and +/- 3 degrees from the ideal position (Fig. 4 (A)). As the internal robot moved through the face, its alignment skis could accommodate up to +/- 1.78 cm offset in-plane of the voxel face it was moving through. When fully extended, these alignment skis enforced an in-plane alignment to <0.1mm (in X and Z) (Fig. 4 (B)). As the internal robot grippers extended, the alignment claws could accommodate an offset of up to 1.34 cm out-of-plane, pulling the two voxels together to enforce an alignment to <0.1mm at full grip (Fig. 4 (C)). Final preload between the faces was applied by the actuation of the fastener, which could accommodate misalignment up to 0.2 cm (in Y). Fastener tensioning resulted in preloading of alignment features on the voxel faces, which served as a positional coupling with repeatability within thermally induced dimensional variation of the voxel faces in a laboratory environment. These coupling systems were over-constrained to implement elastic averaging within voxel-to-voxel interfaces and across assemblies. This provided global dimensional precision within the aforementioned thermal variation for assemblies of any size.

## Laboratory Construction Test

In a laboratory experiment, two transport robots and one fastening robot autonomously assembled a 256-voxel shelter structure (Fig. 5 (A-C) and Movie 1). An extra copy of the fastening robot was occasionally substituted in to continue autonomous operations while the other was being maintained. The shelter structure was selected to show the generality of the system by highlighting the ability to build overhanging geometries. The build started from a five voxel ‘seed’ structure that served as a starting point for the robots. Voxels were manually loaded onto the cargo holder of a transport robot, which then transported the voxel to the build front. There, the crane transport robot removed the ‘cargo’ voxel from the cargo transport robot and placed it into the proper lattice position. Once in position, the fastening robot performed final alignment and attached the new unit cell to the existing structure. Based on a manually generated build plan consisting of voxel build order and robot path plans, commands were sent wirelessly via a centralized controller and operator interface. A simulator was used to check the build plan for robot and structure collisions. The operators were only present for planned shutdowns and fault monitoring and did not send manual commands (except to retry and restart operations if a fault was triggered). The control system issued primitive commands representing discrete actions, such as ‘step forward’, ‘step up’, or ‘turn right 90 degrees.’ The robots then executed pre-established motion routines (stored locally on the robot) using minimal feedback sensing. On the transport robots, this sensing consisted of foot gripper position limit switches, motor current sensing (yaw and gripper motors), and main drive position encoders. Each joint had an encoder at the motor output and at the joint output after timing belt gearing. On the fastening robot, sensing included a position encoder for central rotation motor position tracking, contracted position limit switch, an inertial measurement unit (IMU), and servo electrical current monitoring. The system did not utilize a motion capture system or any other external reference. Instead, the system used discrete dead reckoning, which, because of structural discretization, provided a consistent and reliable reference system.

The 256-voxel build took a total of 4.2 days of continuous run time to complete, which corresponds to assembly throughput rate of 20,000 mm<sup>3</sup>/s. The time for completing the addition of each voxel is available in fig. S7. Each robot successfully conducted thousands of autonomous operations, including locomoting, placing voxels, and bolting fasteners. Distance traveled varied by robot role, since the fastening and crane transport robot remained primarily at the build front (table S2). The cargo transport robot traveled 4624 body lengths (3.15km), the crane transport robot traveled 522 body lengths (0.356 km), and the fastening robot traveled 754 body lengths (0.230 km). Travel was assessed from the movement of each robot’s center of gravity, calculated for each movement primitive using the robotic mass distributions (tables S3 and S4).

Both robot types were able to monitor their operations and detect faults during the build. Each fault paused operation and notified the operator for further instruction, which could either include remote resolution of the issue (no touch recovery) or a manual touch recovery before restarting autonomous operation. The external robot faults included motor overcurrent, main joint motor driver communication time-out, main joint target timeout (when the motor did not reach its target within expected

time), encoder mismatch fault (belt skip detection between motor and joint encoders), gripper failure fault, and board synchronization faults. The internal robot faults included bolter modules not reaching the required fastening torque, extension modules not fully contracted, rotation module not reaching its target position, robot global orientation error when the robot was manually placed into the structure in the incorrect orientation, motor overcurrent, and controller board communication loss (29). In the analysis of system faults (Table 1), faults were divided into two categories: those due to mechanical issues with the robots and those due to communication loss time-out or tether interference that was unsuccessfully managed by the operators, requiring a pause in operations. Since systems in an operational environment would utilize battery power and operate with a different communication architecture, our fault analysis focused on mechanical faults.

During the experiment, the crane transport robot performed 478 primitive operations, with a total of 10 faults due to mechanical failure; of these faults, 7 necessitated a touch recovery (98.5% success rate). The cargo transport robot performed 14,185 primitive operations, with 218 yielding faults from mechanical failure; of these faults, 118 necessitated a touch recovery (99.2% success rate). Many faults were able to be recovered simply by retrying the failed operation. Many of the failures could be attributed to mechanical failure of a support strut on the chassis due to fatigue from a small radius feature and belt skipping due to wear on the main joint. Both issues have been addressed in subsequent robot designs. Many of the gripper failures could be attributed to failed locomotion due the failed support strut or to the location of the gripper sensors, which lead to several incorrect grip detections. These could often be recovered without touch intervention by simply retrying the gripping operation. We expect that this issue can be addressed in subsequent robot design iterations to refine sensor placement.

The internal robot took 711 steps (re-orientations between faces) and bolted 596 voxel faces together. The internal robot experienced 46 locomotion faults due to mechanical failure, requiring 30 touch recoveries (95.8% success rate). The internal robot experienced 33 bolting faults due to mechanical failure, of these faults, 7 required a touch recovery (98.7% success rate). Most of the locomotion faults were due to the extension module not seating well in the contracted position. The bolting faults were due to slight misalignment between the voxel that was placed and the existing structure (which was not able to be captured by the alignment feature envelopes on the robot). The bolting faults were frequently coupled with gripper faults, and many bolter faults were able to be remedied by retrying the gripping motion to seat the bolter modules correctly over the fasteners. Most of these faults were fixed by a no-touch recovery simply by retrying the operation (system paused and notified the operator, followed by operator telling the system to try again and proceed). Increased reliability could be achieved by refining gripping sensing or slightly increasing the gripping capture envelope.

### **Disassembly and Reconfiguration**

The system also demonstrated disassembly and reconfiguration (Fig. 5 (D) and Movie 1). The fastening robot could unbolt fasteners between voxels, which allowed a transport robot to remove the voxel and place it into position elsewhere for reattachment. This reconfiguration is key for not only the re-programmability of the structure and adaptability, but also to the scalability and ability to perform error

correction and discrete repair. Previous work showed that discrete lattices can regain their performance after damage by replacing broken unit cells (35), which forms the most basic concept of operations for the system to perform repair or recover from errors. During early development testing of the robots, robots demonstrated the ability to manipulate broken voxels (due to redundant alignment and gripping features) and even locomoted on voxels with broken struts. Full implementation of repair and system health monitoring would be integrated through the addition of sensors to the robots or voxels.

### **Autonomous Multi-Crew Path Planning and Assembly Simulation**

The system allowed for autonomous generation of a build plan based on the input of a desired geometry of the target structure and the order of its construction. Given a sequence of coordinates of voxels, a corresponding sequence of robot paths can be planned using an algorithm that combines cooperative A\* (36), multi-labeled A\* (37), and an approach to ensure the system avoids locked configurations.

The discrete nature of the structure allowed for a graph-based representation of the operating environment of the robots. For example, for the surface robots, each valid pose of a robot on the surface of a structure corresponded to a node in the state graph. Two nodes were connected with an undirected edge if the robot could transition between the two corresponding poses in one move. A path of a robot from its initial configuration to any desired configuration corresponded to a path in the graph. To avoid robot collisions, we introduced a time-component to the nodes, and allowed the robots to reserve free nodes when planning their paths. Note that both graphs changed in time as the voxel structure changed, and these changes were computed along with the path planning computation.

The overview of the path planning algorithm is the following. The paths for each robot were computed iteratively. For each next voxel to be attached to the structure, a valid tuple  $(p_f, p_w, p_b)$  of hand-off poses for the cargo and crane robots was computed, where  $p_w$  was the pose in which the crane waited for the cargo robot to bring the voxel,  $p_f$  was the pose in which the cargo fed the voxel to the crane, and  $p_b$  was the pose in which the crane placed the voxel. Pose  $p_w$  was positioned such that in one motion the crane robot could pick up the cargo voxel carried by a cargo robot in pose  $p_f$ , and place it using the pose  $p_b$  (to be attached by the interior fastening robot to the structure). At every iteration, the robots planned their paths from their current poses to the location of the assembly of the next voxel. The cargo robots planned their paths from their current location to the depot node  $d$  (the pose in which they received a cargo voxel at the depot), and from  $d$  to the next pose feeding the corresponding crane robot.

To ensure feasibility of a solution, we utilized a multi-labeled A\* approach in path planning for cargo robots. Nodes of the state graph were labeled with a flag denoting whether a cargo voxel had been picked up from the depot. Furthermore, to avoid deadlocks, each robot was assigned a storage location. We ensured that at each step all robots had free paths to their storage locations, and thus the system could never enter a locked state.

Movie S1 shows a video of a simulated build of a large tower by three teams of robots, implementing the autonomous path planning algorithm. The construction

initiated from a base layer of voxels. The robots started from their designated base positions, built the tower, and returned to their bases afterwards. For clarity each team was shown in a different color. The crane external robots assumed build-ready positions and waited for the cargo robots to bring voxels from the depot, which was located at the three protruding voxels in the base layer of the construction. The surface of the construction obscured the operation of the internal robots; therefore, internal robot planning was not demonstrated in this video (but would utilize the same algorithm as the transport robots).

### Structure Mechanical Performance

Both specific strength and specific stiffness are key structural metrics for high performance infrastructure applications. Assemblies of  $3 \times 3 \times 3$  voxel (Fig. 6 (A)) cubes were tested in compression with a mechanical testing machine to evaluate strength and stiffness (Fig. 6 (B)). Failure of the structure occurred at the fastener (Fig. 6 (C)) before buckling of the struts, suggesting that design iterations to strengthen the intervoxel connection could further increase strength. The resulting structure had a strength of 11.17 kPa (+/- 0.77 kPa), corresponding to an average failure load of over 9000 N. This exceeded estimated strengths for many previously demonstrated reconfigurable programmable matter systems at orders of magnitude less density (0.0103 – 0.0121 g/cc, depending on amortization of the robot mass). Figure 6 (D) compares current structural strength per material density with other published self-reprogrammable systems that reported enough information to estimate assembly strength values (no systems were found that directly reported assembled material strength). Estimations were made using methodologies that provide upper bounds for strength (see Materials and Methods section and fig. S8). The structural system had a stiffness of 1.1129 MPa (+/- 0.0430 MPa), which at ultra-light density, achieves a relative stiffness regime typical of high-performance carbon fiber lattice truss systems for space applications. Since no similar reconfigurable robotic systems reported assembly stiffness for comparison, Fig. 6 (E) compares the relative compressive modulus of this work to several other architected lattices (35, 38–41) and a carbon-fiber space truss (42).

The stiffness of the structure,  $E$ , was competitive for high-performance applications at an ultra-light density,  $\rho$ . For many applications, including those typical for space structures, minimum mass performance indices do not scale directly with specific stiffness ( $E/\rho$ ), but rather with  $E^{1/2}/\rho$  or  $E^{1/3}/\rho$  (43). This is the basis of efficiency of trusses and lattice materials. By exceeding the ideal quadratic performance (Fig. 6 (E)), we can assert that the structural system demonstrated linear scaling of relative modulus with relative density (44) for the given loading condition. Lake et al. show that the performance of a precision telescope support is governed by its vibrational modes and that the fundamental frequency of a truss support structure is well approximated by abstracting the truss to a plate made of a lattice material (45). Using rationale and methodology developed in (45), we estimated the bulk material stiffness of historical examples of carbon-fiber truss structures for space applications using the truss size, mass, and first fundamental frequency. For example, a 4m diameter truss aperture (42) made from high performance carbon fiber struts with a mass of 85.7 kg, depth of 0.5m and a first fundamental frequency of 35 Hz could be estimated to have a material stiffness of 0.012 GPa and a volume density of 0.0136 g/cc. Normalizing with the reported strut stiffness of 16.9e6 psi (116.5 GPa) yielded a relative density of 1.02e-4. Further

examples of this rationale and simulations showing the accuracy of this analytical method are available in (46). Importantly, for any given application with specific absolute strength or stiffness requirements, architected lattice literature has demonstrated how the dimensions and constituent materials of lattices of various geometries can be tuned to achieve a wide variety of material properties. Based on architected lattice theory, these results suggest that at ultra-light density, this structural system could achieve a stiffness regime typical of high-performance carbon fiber space trusses given a continuous fiber constituent material (40).

## DISCUSSION

This work demonstrated implementation of programmable matter with material performance that is competitive for high-performance engineering applications, offering autonomous construction and reconfiguration of materials into useful form factors.

For this system demonstration, injection molded chopped fiber composites offered an economical way to manufacture voxels that achieved performance regimes useful for space structures. However, many opportunities exist to optimize the joints and materials used to further extend the structural performance of the system. Previous work on assembled architected lattices show that increased performance can be reached by using higher-performance constituent materials like continuous carbon fiber (38, 40, 43, 47). Additionally, though the chosen voxel geometry and cuboctahedron lattice has a favorable combination of stiffness and features that make it simpler to assemble (including generous clearances for robotic end effectors) (44, 48), we do not claim it is optimal across all possible unit cell geometries. We believe that similar systems can leverage the advantages of relative robotics with different unit cell geometries, or even combine building block geometries (32).

These low-cost and relatively imprecise robots achieved assembly and reconfiguration with very high repeatability and consistency, with minimal required state estimation resources (sensors and computation). The low-cost nature of the robots offers a potential for highly parallel and redundant robotic agents. In addition to increased throughput, this allows for increased reliability, since algorithmic solutions can leverage operational margins in quantities of robots and structural components to work around individual component failures. Also, faults in the system described in this work are attributable to design issues that are natural to uncover during high cycle testing, such as component wear or fatigue failure. We expect these can be mitigated with design refinement. Future implementations of this type of system may still benefit from integration of additional sensing to further increase reliability, add various types of system health monitoring, or to achieve other system level utility. Fully distributed planning and scheduling algorithms can achieve diverse functionality with modest local sensing and communication for regular and discretized systems (such as this system).

Our system meets conventional definitions of programmable matter given the discrete nature, exclusive reliance on local metrology, reconfigurability, and ability to implement relevant algorithms. However, given the structural performance, it is also natural to compare the system to conventional approaches to

robotic assembly. Although traditional robotic systems, like robot arms, have demonstrated great utility in industry, scaling them up is typically considered costly. Since they require highly structured workspaces, robot arm systems are difficult to generalize for and coordinate in unstructured environments. Coordinated robotics in structured environments, on the other hand, is already employed in industry. Challenges of conventional approaches (managing global and local reference frames, alignment with computer vision, metrology, autonomy integrating path planning in an unstructured environment (49)) do not apply to coordinated robotics in structured environments (self-reconfigurable systems). The discrete nature, reliance on local metrology, and algorithmic simplicity, can provide powerful simplification in robotic, sensing, and computational complexity of reliably meeting a given construction goal.

Our system delivers high mechanical performance along with competitive overall system engineering-relevant metrics. Throughput with a single robot team (20,000 mm<sup>3</sup>/s) exceeds that of typical 3D printing techniques (15). This throughput can be increased via parallelism and hierarchical assembly (15, 50), both of which can be readily accommodated by available build algorithms (51, 52). Though this system was demonstrated with a centralized control architecture, for larger systems, distributed control architectures can be more efficient and have been demonstrated in simulation (51). Reconfiguration and disassembly capability allows any target geometry to be built and provides reliable error correction and repair mechanisms that are critical for scalable build sizes and maintainable systems (35). Voxels and fasteners are suitable for mass production at low cost (\$7.43 per face at prototype quantities). By incorporating voxel types with different geometries and/or materials, diverse functionality can be achieved via hierarchy (46) and a small set of part types (32), mirroring the strategy used by digital electronics and biology to achieve scalability, adaptability, error correction, and healing.

Programmable material systems promise versatility, robustness, and low cost via economies of scale, re-use, and generality (7). A hierarchical ‘building blocks’ approach, as understood to be proven scalable by biology, provides the strategy for solving problems of large systems. Bringing this functionality into the ultra-light, high-performance structural regime, and solving the ‘big systems’ problem (7), enables new applications in infrastructure, aerospace, and exploration. The effect of revolutionizing engineered material lifecycles in everyday infrastructure applications could transform the way we live, build, and adapt to a changing climate (2).

## **MATERIALS AND METHODS**

### **Structure Materials and Testing**

Each fastener and square voxel face (fig. S1) was injection molded from a commercial 40% chopped carbon fiber reinforced polyamide compound (StattechNN-40CF). Standard geometry tension test coupons were also injection molded and tested on a universal testing machine in tension at 0.1 mm/min. Since the injection molded faces have a knit line at the center of each strut, the test coupons were injection molded such that there were both specimens with knit lines (Type I and Type II) and without knit lines (Type III) (fig. S2). Type III data will be used to normalize the lattice performance for relative stiffness metrics, since it represents the ideal material performance, and the presence of knit lines is seen as a

penalty of the chosen manufacturing process and face design. Modulus was calculated using a common auto-modulus algorithm that divides the data into six sections, performs a linear regression on each section, and selects the highest modulus section. The results of six specimens are in table S5 and stress-strain curves are shown in figure S4. Ultimate strength was only recorded if the specimen broke within the gauge length of the extensometer. Type I and II data is reported in table S6 to give an understanding of the performance of the lattice relative to the actual as-manufactured material performance. This provides information necessary to understand the performance improvements that can be realized with the utilization of higher-performance constituent materials or different manufacturing processes.

For fastener tensile strength testing, fastener pairs were loaded into a custom fixture (fig. S5) and actuated (turned) so that the fastener teeth were fully engaged, as in a lattice assembly. Fasteners were then pulled in tension until failure (12mm/min extension rate). Table S1 shows the break load of each fastener. Note, since the fastener is designed to preload the joint of the inter-voxel connection, this value should not be understood as the separation force between two voxels (which is higher, as is characteristic of typical pre-loaded joints).

The square injection molded faces are assembled into voxels using 10-32 stainless steel bolts and nuts at the corners of the faces. Three 3x3x3 voxel assembly specimens were hand-assembled using manually actuated fasteners. The specimens were cycled between 0 and 1500N compressive force at 36 mm/min, then compressed to failure at the same extension rate. The sample was not fixtured to the compression plates. An initial load-up portion of the curve can be observed where the mass of the compressive plate (980 N) is taken up and a spherical seat connecting this plate to the load cell is properly loaded. All curves are zeroed in relation to this pre-load and modulus measurements taken after this pre-load (which represents even loading of the lattice). Failure occurred at the root of the fastener tooth, which was predicted by FEA analysis of the fastener. All failures were on the center voxel column. Ultimate strength was calculated from the maximum load. A chord modulus is reported during the consistent loading phase of the stress strain curve, which is conservative compared to the auto-modulus algorithm previously described (table S7). Since previous experiments show that the stiffness of the fixture is more than an order of magnitude stiffer than the sample, a fixture compliance correction was not applied to the data. The relative stiffness value was calculated by normalizing the average chord modulus with the average non-knit constituent material stiffness (to report the most conservative value). Calculations of voxel density based on component mass measurements are shown in table S8.

### **Robot System Architecture**

Minimal robot capabilities for shape universal programmable matter are described in the literature, including for the lattice geometry employed in this study (52). These capabilities are a subset of the capabilities of the robots employed in this study. We use more capable robots for the simple reason that available commercial off-the-shelf actuators and motion control components provided additional capability when selected to meet the minimum requirements. It is expected that the cost and performance of this kind of system can be improved even further with development of optimized systems with high force/torque density but low necessary control bandwidth.

## Transport Robot

The exterior transport robot used in this system is named Scaling Omnidirectional Lattice Locomoting Explorer (SOLL-E) and is an inchworm style bipedal robot that locomotes on the exterior of the lattice structure (33). The robot has three main joints, each driven by a brushless motor (SunnySky M8 BLDC) and timing belt for gearing. Controls are handled with three custom onboard controller boards incorporating simple system on chip (SoC) microcontrollers (ESP32 modules), two of which are paired with modular commercial off-the-shelf BLDC controllers (ODrive V3.6, one of which is utilized to control two of the main drives). The robot has two feet, both of which have grippers that can grasp the structure (using four Hitec D89MW servos per foot), alignment features to ensure proper positioning, and “yaw” modules that allow basal plane rotation of the foot (driven by a RobotZone 26 RPM DC gearmotor). Each rotational degree of freedom has an absolute position encoder at the joint output (after gearing) and a relative encoder at the motor output for motor control. Each gripper has a position limit switch to detect successful gripping of the structure. To allow the robot to carry unit cells, one side of the robot is fitted with a ‘cargo voxel gripper,’ which is a copy of the alignment features and grippers on the robot feet. The placement ‘crane’ role starts with positioning at a location and using one of its foot grippers to grab a delivered voxel from the ‘cargo’ robot. The robot acting in the ‘crane’ role then locomotes to place this voxel in a desired location. It is capable of picking and placing from and to either ‘cargo’ or a position on a lattice, and is thereby capable of adding, removing, and repositioning voxels.

The robot linkage (‘leg’) geometry is optimized for minimization of required work with key locomotion functionality. The ‘reach-over’ bend in the main leg struts improves the work-cost function by allowing the legs to clear stepping entirely over a voxel avoiding collisions between the robot and structure. It also provides a suitable cargo gripper position that makes the cargo voxel more accessible for the robot acting as a ‘crane’. State estimation incorporates current sensing, limit switches, and rotary encoder data to manage system process flow. During the ARMADAS ground demonstration, SOLL-E successfully executed about 15,000 actions (steps or placements).

SOLL-E operations are modularized into trajectories that define continuous motion between discrete configurations that are indexed to the structure. The capability of remote control of individual degrees of freedom was implemented for the development of these trajectory modules together with inverse kinematics, and full-scale planning was implemented at the trajectory module level of hierarchy (and higher). Transitions across the discrete state reconfiguration map are pruned for manual and algorithmic planning purposes based on local configuration of structure and other robots. SOLL-E's number and type of allowed motions is dictated by the needs of the assembly. Trajectories, and their waypoints, of the required motions, such as “step forward” or “step up”, are hardcoded in the robot firmware and called by the central control station. Motion coordination among the three control boards of the robot is achieved by moving onto the next waypoint only when all the boards have successfully reached the previous waypoint, within a margin to ensure a smooth motion. By moving between fixed configurations, SOLL-E can efficiently locomote anywhere on the structure and place voxels with a small number of fixed and simple trajectories (that are well tested and validated before a build).

## Fastening Robot

The interior fastening robot used in this system is named Mobile Metamaterial Internal Co-Integrator (MMIC-I) and is an inchworm style robot that moves between adjacent voxel faces in the interior of the lattice structure (29). It joins newly placed voxels to the existing structure by locomoting to the faces of the unit cell to be added and bolting the four fasteners at the corner of each face. The bolting modules simultaneously push axially on the fastener head, and then rotate the fasteners (approximately 60 degrees) to fully lock the two faces together (0.9 Nm running torque to 1.5 Nm recognized as hard stop).

The robot is symmetric to allow it to move forwards and backwards (Fig. 3 and fig. S6). Each half of the robot contains an arm module consisting of four servos (Hitec D980) that power a Sarrus linkage, which extends and contracts to propel the robot forward during a step. Both sides also contain a gripper module that allows the robot to hold onto a voxel face between each stepping motion. It consists of four crank-slider mechanisms each driven by a servo (Hitec D89) that simultaneously extend and align to each node of the voxel face. Each gripper mechanism is outfitted with an alignment claw that engages with the voxel node. As the four grippers extend, the claws slide along the voxel nodes, centering the robot and accounting for small translational and rotational offsets. A rotation module is located in the middle of the robot and allows turning to reorient to any orthogonal direction. It is driven by a servo (Hitec D980) with an integrated absolute encoder for position sensing and control.

The bolter modules are integrated into each mechanism of the gripper subassembly and are positioned into the bolting position when the robot is in a contracted arm position where the grippers are fully engaged with the voxel face. Each bolter module contains two servos (Hitec D951 and Hitec D89) to apply the axial force then rotational torque (respectively) to specification.

The robot is equipped with two custom controller boards and two 11.1 V 1000 mAh LiPo batteries. The controller board contains an ESP32 microcontroller, current sensing circuits for each subassembly, and additional sensing (magnetic absolute encoder, switches, IMU) for system monitoring and fault checking.

## 256 Voxel Demonstration Build

The large-scale build demonstration was conducted under operator supervision using a custom operating software interface. The build plan was uploaded onto the software interface, and the operation software autonomously coordinated all three robots by sending commands and receiving robot status and system health feedback during the build. The user interface logged and displayed real time data from the robots (for both data collection and fault monitoring/recovery). The SOLL-E robots recorded main joint position, foot yaw rotation angle, gripper open/close position, and status feedback for macro trajectories. The MMIC-I robot provided servo position data, hip rotation angle, gripper open/close position, servo current draw, battery voltage, and status feedback for locomotion primitive trajectories and bolter commands.

The robots were shut down every evening and restarted during business hours. The two SOLL-E robots ran on tethered power and MMIC-I ran on batteries that were periodically manually replaced throughout the build as needed (build sequence paused when low battery indicated, battery was manually swapped, build sequence resumed). A single MMIC-I battery set could operate the robot for approximately three hours, depending on the number of motions that were completed.

The timelapse video shows the full 256 voxel build. The three robots started the build on a seed structure consisting of five seed voxels. The voxels were loaded onto the cargo SOLL-E at the beginning of the seed structure for transport to the build front. The robots did not receive manual assistance to complete any part of the build, although they were allowed to re-attempt parts of the build after recovering from a fault or receiving maintenance (including replacement of worn or broken components or robot instances for repair, the need for which has been addressed in further robot versions). If a robot fault was triggered, the robot autonomously performed a fault recovery routine and retried the command. If the robot was unable to complete self-recovery from a fault, the operation software would automatically pause the build sequence, and the robots were reset to their starting positions to retry the command and continue the build.

The cumulative build time for the 256-voxel structure is approximately 100.4 hours or 4.2 days. This value is the time it would take for the robots to assemble the structure without any pauses or stoppages. Figure S4 shows the elapsed time it takes to join each voxel to the structure. The variation in time depends on the distance the voxel needs to be transported from the voxel pick up location. Optimizations in build time for future work could come from increased robot speed, more efficient path planning, multiple feed stations, parallelization, and more (50). Build throughput is calculated as the volume of the 256 voxels divided by the total build time.

### **Robot Distance Traveled**

Comparing the performance of robots at different scales is a nuanced task, but one metric that can be used is the “Distance Traveled per Robot Body Length”. This metric normalizes the distance traveled for a robot by dividing the total distance traveled by the robot body length. In this system, these values are presented for each robot in table S2.

To calculate the total distance traveled for each robot, we summed the center of gravity (CG) displacement for each primitive motion in the complete 256 voxel structure build plan. Each primitive has a start and end configuration, and distance between the CG for each configuration gives the displacement. These values were calculated using a CAD model with approximate mass distributions and does not account for variations from voxel payloads. The mass breakdown for each assembly used for distance traveled calculations are shown in tables S3 and S4.

### **Reconfigurable Robot Literature Example Strength Estimates**

Since no literature examples of structural programmable matter reported the strength of the resulting matter, estimations were made using provided system data. Not all systems described in the literature provided enough data to make an estimation. For each comparison, every effort was made to estimate the strength in a manner that would bias towards overestimating the resulting strength.

The Rigid Body Tension estimation method assumes that simple cubic packing modules are perfectly rigid, and that the failure method will be the connection between the modules (fig. S8 (A)). Therefore, the strength is the unit separation force divided by the unit cross-sectional area. The unit separation force between modules was consistently reported by many published systems and provided the best method for estimating system strength. Since these modules are not perfectly rigid, we can expect this estimation to be a generous upper bound.

For a reconfigurable truss system, the rigid body tension estimation needed to be modified. For these systems (28), we assume an analogously sized lattice structure to the experimentally tested 3x3x3 voxel structure. In the case of reconfigurable strut and nodes, this is a cubic lattice three strut lengths wide. Given the size (maximum) of the strut and the strength of the strut, we can estimate a similar rigid body tensile strength (fig. S8 (B)). We were only able to find one published reconfigurable lattice system with enough reported information to estimate system strength.

For a small number of systems, enough information was given to give a better estimation of the strength. For the material-robot system presented by Jenett et al. (15), since the elements are also face-connected cuboctahedron, an assumption of fastener limited cuboctahedron lattice scaling can be applied. In this case, the separation force is estimated from the size of magnets, which was estimated as 55N +/- 5N. Since the relative densities of the two lattices are similar and both are fastener limited, it is reasonable to assume that the maximum load will scale with maximum fastener strength. In reality, the material used by the material-robot is less stiff than the lattice of the current work, so strut-based failure might occur earlier. Therefore, this estimate should still be conservative. Note that for a cuboctahedron lattice, the compressive strength is expected to be approximately half of the tensile strength (35).

## Supplementary Materials

Figs. S1 to S8  
Tables S1 to S9  
Movie S1

## References and Notes

1. J. Von Neumann, *Theory of Self-Reproducing Automata* (University of Illinois Press, 1966).
2. E. G. Hertwich, Increased carbon footprint of materials production driven by rise in investments. *Nat. Geosci.* **14**, 151–155 (2021).
3. A. Bergamini, T. Delpero, L. D. Simoni, L. D. Lillo, M. Ruzzene, P. Ermanni, Phononic Crystal with Adaptive Connectivity. *Adv. Mater.* **26**, 1343–1347 (2014).
4. S. G. Carrillo, L. Trimby, Y. Au, V. K. Nagareddy, G. Rodriguez-Hernandez, P. Hosseini, C. Ríos, H. Bhaskaran, C. D. Wright, A Nonvolatile Phase-Change Metamaterial Color Display. *Adv. Optical Mater.* **7**, 1801782 (2019).
5. E. Hawkes, B. An, N. M. Benbernou, H. Tanaka, S. Kim, E. D. Demaine, D. Rus, R. J. Wood, Programmable matter by folding. *Proc Natl Acad Sci USA.* **107**, 12441–12445 (2010).
6. M. R. Jones, N. C. Seeman, C. A. Mirkin, Programmable materials and the nature of the DNA bond. *Science.* **347**, 1260901–1260901 (2015).
7. M. Yim, W.-M. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, G. S. Chirikjian, Modular Self-Reconfigurable Robot Systems, 10 (2007).
8. W. McCarthy, Programmable matter: How quantum wellstone ushered in our modern age. *Nature.* **407**, 1 (2000).

9. B. Florijn, C. Coullais, M. van Hecke, Programmable Mechanical Metamaterials. *Phys. Rev. Lett.* **113**, 175503 (2014).
10. S. M. Felton, M. T. Tolley, R. J. Wood, "Mechanically programmed self-folding at the millimeter scale" in *2014 IEEE International Conference on Automation Science and Engineering (CASE)* (IEEE, New Taipei, Taiwan, 2014; <http://ieeexplore.ieee.org/document/6899484/>), pp. 1232–1237.
11. M. T. Tolley, S. M. Felton, S. Miyashita, D. Aukes, D. Rus, R. J. Wood, Self-folding origami: shape memory composites activated by uniform heating. *Smart Mater. Struct.* **23**, 094006 (2014).
12. G. M. Whitesides, B. Grzybowski, Self-Assembly at All Scales. *Science*. **295**, 2418–2421 (2002).
13. P. Flocchini, G. Prencipe, N. Santoro, Eds., *Distributed Computing by Mobile Entities: Current Research in Moving and Computing* (Springer International Publishing, Cham, 2019; <http://link.springer.com/10.1007/978-3-030-11072-7>), vol. 11340 of *Lecture Notes in Computer Science*.
14. S. S. R. Chennareddy, A. Agrawal, A. Karuppiah, Modular Self-Reconfigurable Robotic Systems: A Survey on Hardware Architectures. *Journal of Robotics*. **2017**, 1–19 (2017).
15. B. Jenett, A. Abdel-Rahman, K. Cheung, N. Gershenfeld, Material–Robot System for Assembly of Discrete Cellular Structures. *IEEE Robot. Autom. Lett.* **4**, 4019–4026 (2019).
16. G. Liang, H. Luo, M. Li, H. Qian, T. L. Lam, "FreeBOT: A Freeform Modular Self-reconfigurable Robot with Arbitrary Connection Point - Design and Implementation" in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, Las Vegas, NV, USA, 2020; <https://ieeexplore.ieee.org/document/9341129/>), pp. 6506–6513.
17. Y. Tu, G. Liang, T. L. Lam, "FreeSN: A Freeform Strut-node Structured Modular Self-reconfigurable Robot - Design and Implementation" in *2022 International Conference on Robotics and Automation (ICRA)* (IEEE, Philadelphia, PA, USA, 2022; <https://ieeexplore.ieee.org/document/9811583/>), pp. 4239–4245.
18. S. Hauser, M. Mutlu, P.-A. Léziart, H. Khodr, A. Bernardino, A. J. Ijspeert, 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**, 103467 (2020).
19. J. W. Romanishin, K. Gilpin, D. Rus, "M-blocks: Momentum-driven, magnetic modular robots" in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, Tokyo, 2013; <http://ieeexplore.ieee.org/document/6696971/>), pp. 4288–4295.
20. V. Zykov, A. Chan, H. Lipson, "Molecubes: An Open-Source Modular Robotics Kit" in *IROS-2007 Self-Reconfigurable Robotics Workshop* (2007).
21. J. Neubert, H. Lipson, Soldercubes: a self-soldering self-reconfiguring modular robot system. *Auton Robot.* **40**, 139–158 (2016).
22. A. Spinos, D. Carroll, T. Kientz, M. Yim, "Variable topology truss: Design and analysis" in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, Vancouver, BC, 2017; <http://ieeexplore.ieee.org/document/8206098/>), pp. 2717–2722.
23. P. Swisler, M. Rubenstein, "FireAnt3D: a 3D self-climbing robot towards non-latticed robotic self-assembly" in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, Las Vegas, NV, USA, 2020; <https://ieeexplore.ieee.org/document/9341116/>), pp. 3340–3347.
24. Y. Yoon, D. Rus, "Shady3D: A Robot that Climbs 3D Trusses" in *Proceedings 2007 IEEE International Conference on Robotics and Automation* (IEEE, Rome, Italy, 2007; <http://ieeexplore.ieee.org/document/4209722/>), pp. 4071–4076.

25. D. Hjelle, H. Lipson, "A robotically reconfigurable truss" in *2009 ASME/IFTOMM International Conference on Reconfigurable Mechanisms and Robots* (London, UK, 09), pp. 73–78.
26. Seung-kook Yun, D. A. Hjelle, E. Schweikardt, H. Lipson, D. Rus, "Planning the reconfiguration of grounded truss structures with truss climbing robots that carry truss elements" in *2009 IEEE International Conference on Robotics and Automation* (IEEE, Kobe, 2009; <http://ieeexplore.ieee.org/document/5152714/>), pp. 1327–1333.
27. N. Melenbrink, P. Michalatos, P. Kassabian, J. Werfel, "Using local force measurements to guide construction by distributed climbing robots" in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, Vancouver, BC, 2017; <http://ieeexplore.ieee.org/document/8206298/>), pp. 4333–4340.
28. A. Lyder, R. Garcia, K. Stoy, "Mechanical design of odin, an extendable heterogeneous deformable modular robot" in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, Nice, 2008; <http://ieeexplore.ieee.org/document/4650888/>), pp. 883–888.
29. O. Formoso, G. Trinh, D. Catanoso, I.-W. Park, C. Gregg, K. Cheung, "MMIC-I: A Robotic Platform for Assembly Integration and Internal Locomotion through Mechanical Meta-Material Structures" in *2023 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, London, United Kingdom, 2023; <https://ieeexplore.ieee.org/document/10161263/>), pp. 7303–7309.
30. I.-W. Park, D. Catanoso, O. Formoso, C. Gregg, M. Ochalek, T. Olatunde, F. Sebastianelli, P. Spino, E. Taylor, G. Trinh, K. Cheung, "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, 2023), p. (In Press).
31. M. Carney, B. Jenett, "Relative Robots: Scaling Automated Assembly of Discrete Cellular Lattices" in *Proceedings of the ASME 2016 11th International Manufacturing Science and Engineering Conference. Volume 2: Materials; Biomanufacturing; Properties, Applications and Systems; Sustainable Manufacturing* (American Society of Mechanical Engineers, Blacksburg, Virginia, USA, 2016; <https://asmedigitalcollection.asme.org/MSEC/proceedings/MSEC2016/49903/Blacksburg,%20Virginia,%20USA/268940>).
32. B. Jenett, C. Cameron, F. Tournomousis, A. P. Rubio, M. Ochalek, N. Gershenfeld, Discretely assembled mechanical metamaterials. *Sci. Adv.* **6**, eabc9943 (2020).
33. N. B. Cramer, D. W. Cellucci, O. B. Formoso, C. E. Gregg, B. E. Jenett, J. H. Kim, M. Lendraitis, S. S. Swei, G. T. Trinh, K. V. Trinh, K. C. Cheung, Elastic shape morphing of ultralight structures by programmable assembly. *Smart Mater. Struct.* **28**, 055006 (2019).
34. O. Formoso, C. Gregg, G. Trinh, A. Rogg, K. Cheung, "Androgynous Fasteners for Robotic Structural Assembly" in *2020 IEEE Aerospace Conference* (IEEE, Big Sky, MT, USA, 2020; <https://ieeexplore.ieee.org/document/9172583/>), pp. 1–8.
35. C. E. Gregg, J. H. Kim, K. C. Cheung, Ultra-Light and Scalable Composite Lattice Materials. *Adv. Eng. Mater.* **20**, 1800213 (2018).
36. D. Silver, Cooperative Pathfinding. *AIIDE*. **1**, 117–122 (2021).
37. F. Grenouilleau, W.-J. V. Hoeve, J. N. Hooker, A Multi-Label A\* Algorithm for Multi-Agent Pathfinding. *ICAPS*. **29**, 181–185 (2021).
38. L. Dong, H. Wadley, Mechanical properties of carbon fiber composite octet-truss lattice structures. *Composites Science and Technology*. **119**, 26–33 (2015).
39. X. Zheng, H. Lee, T. H. Weisgraber, M. Shusteff, J. DeOtte, E. B. Duoss, J. D. Kuntz, M. M. Biener, Q. Ge, J. A. Jackson, S. O. Kucheyev, N. X. Fang, C. M. Spadaccini, Ultralight, ultrastiff mechanical metamaterials. *Science*. **344**, 1373–1377 (2014).

40. K. C. Cheung, N. Gershenfeld, Reversibly Assembled Cellular Composite Materials. *Science*. **341**, 1219–1221 (2013).
41. T. A. Schaedler, A. J. Jacobsen, A. Torrents, A. E. Sorensen, J. Lian, J. R. Greer, L. Valdevit, W. B. Carter, Ultralight Metallic Microlattices. *Science*. **334**, 962–965 (2011).
42. H. G. Bush, C. L. Herstrom, W. L. Jr. Heard, T. J. Collins, W. B. Fichter, R. E. Wallsomt, J. E. Phelps, Design and Fabrication of an Erectable Truss for Precision Segmented Reflector Application. *J. Spacecraft*. **28**, 251–257 (1991).
43. S. M. Arnold, D. Cebon, M. Ashby, “Materials Selection for Aerospace Systems” (2012–217411, NASA, Reston ,VA, 2018).
44. W. Chen, S. Watts, J. A. Jackson, W. L. Smith, D. A. Tortorelli, C. M. Spadaccini, Stiff isotropic lattices beyond the Maxwell criterion. *Sci. Adv.* **5**, eaaw1937 (2019).
45. M. S. Lake, L. D. Peterson, M. B. Levine, Rationale for Defining Structural Requirements for Large Space Telescopes. *Journal of Spacecraft and Rockets*. **39**, 674–681 (2002).
46. B. Jenett, C. Gregg, D. Cellucci, K. Cheung, "Design of multifunctional hierarchical space structures" in *2017 IEEE Aerospace Conference* (IEEE, Big Sky, MT, USA, 2017; <http://ieeexplore.ieee.org/document/7943913/>), pp. 1–10.
47. M. F. Ashby, R. F. M. Medalist, The mechanical properties of cellular solids. *MTA*. **14**, 1755–1769 (1983).
48. M. Ochalek, B. Jenett, O. Formoso, C. Gregg, G. Trinh, K. Cheung, "Geometry Systems for Lattice-Based Reconfigurable Space Structures" in *2019 IEEE Aerospace Conference* (IEEE, Big Sky, MT, USA, 2019; <https://ieeexplore.ieee.org/document/8742178/>), pp. 1–10.
49. K. H. Petersen, N. Napp, R. Stuart-Smith, D. Rus, M. Kovac, A review of collective robotic construction. *Sci. Robot.* **4**, eaau8479 (2019).
50. A. Abdel-Rahman, C. Cameron, B. Jenett, M. Smith, N. Gershenfeld, Self-replicating hierarchical modular robotic swarms. *Commun Eng.* **1**, 35 (2022).
51. A. Costa, A. Abdel-Rahman, B. Jenett, N. Gershenfeld, I. Kostitsyna, K. Cheung, "Algorithmic Approaches to Reconfigurable Assembly Systems" in *2019 IEEE Aerospace Conference* (IEEE, Big Sky, MT, USA, 2019; <https://ieeexplore.ieee.org/document/8741572/>), pp. 1–8.
52. E. Niehs, A. Schmidt, C. Scheffer, D. E. Biediger, M. Yannuzzi, B. Jenett, A. Abdel-Rahman, K. C. Cheung, A. T. Becker, S. P. Fekete, "Recognition and Reconfiguration of Lattice-Based Cellular Structures by Simple Robots" in *2020 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, Paris, France, 2020; <https://ieeexplore.ieee.org/document/9196700/>), pp. 8252–8259.

## Acknowledgments:

**Funding:** NASA Space Technology Mission Directorate, Game Changing Development Program (Automated Reconfigurable Mission Adaptive Digital Assembly Systems – ARMADAS – Project)

### **Author contributions:**

Conceptualization: KC, CG, OF, GT, IP

Methodology: KC, CG, OF, GT, DC, TO, IP

Material and Structural Characterization: KC, OF, CG

Robotic Development and Demonstration: KC, CG, OF, GT, DC, TO, IP, MO

Planning, Scheduling, Simulation, and Algorithms: KC, OF, CG

Funding acquisition: KC, ET

Project administration: CG, KC, ET

Supervision: KC, ET

Writing – original draft: CG, KC

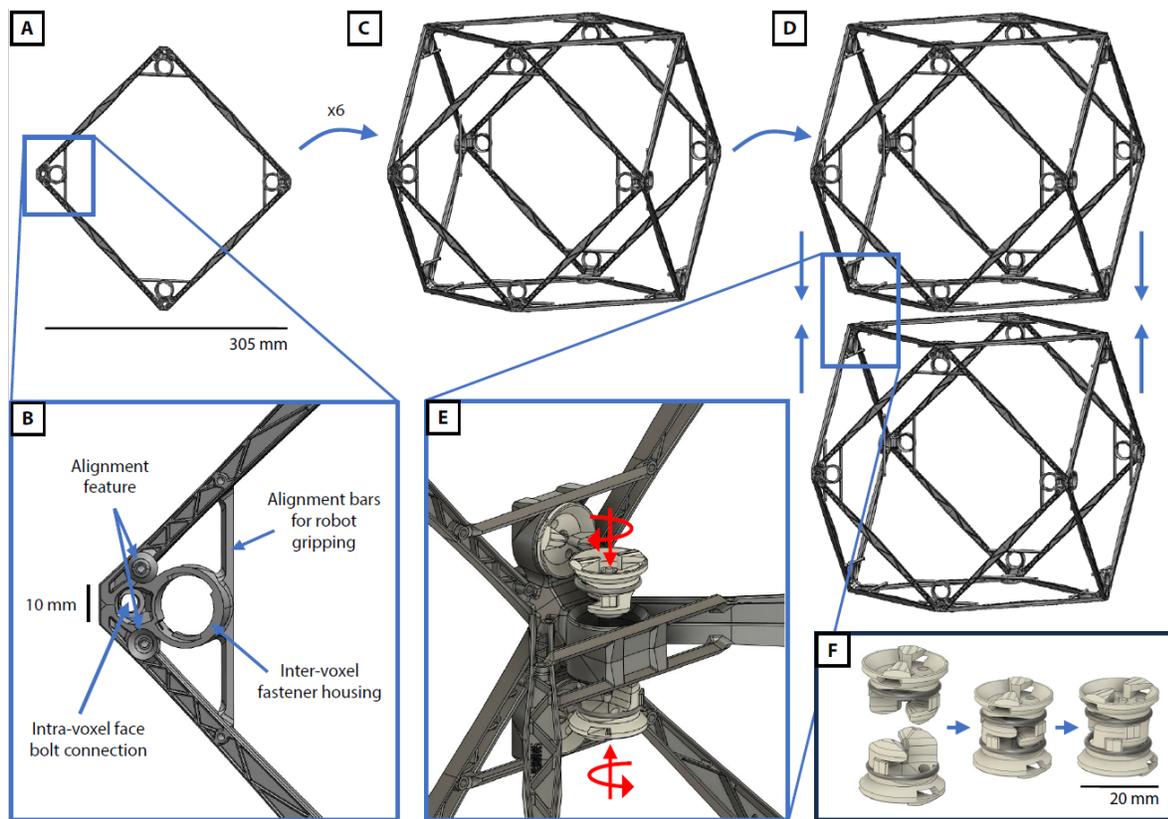
Writing – original draft-supplementary: CG, OF, DC, GT

Writing – review & editing: CG, OF, GT, KC, DC, MO, IK, FS

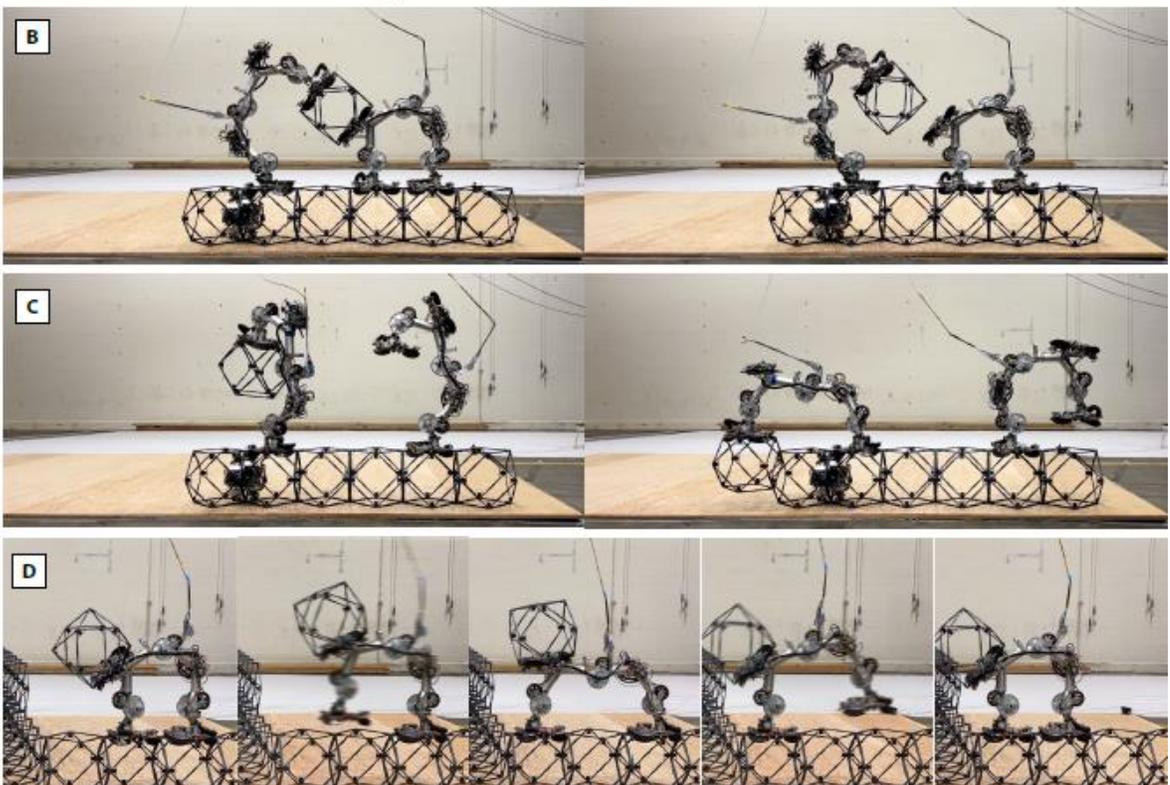
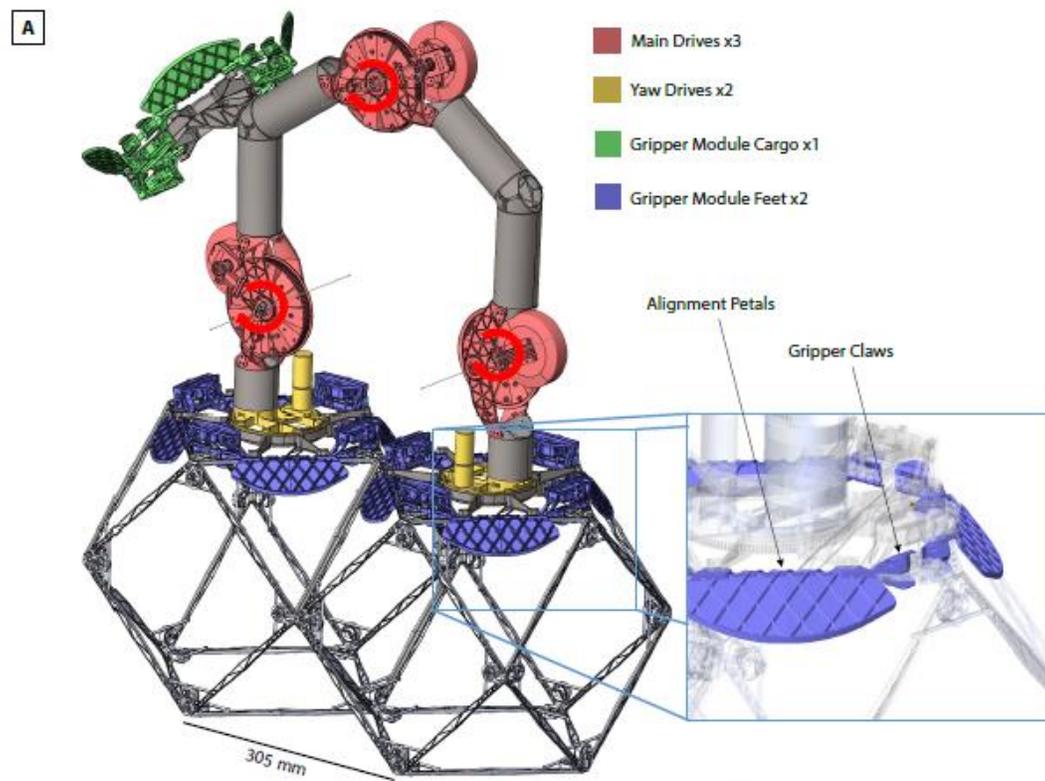
**Competing interests:** Patent applications have been filed on the design of transport and fastening robots that operate on the exterior and interior of periodic structures, as well as the design and manufacturing of modular structural components.

**Data and materials availability:** All summary data necessary to reproduce these results are available in the main text or the supplementary materials. Detailed data files are available in the repository (DOI: 10.5061/dryad.3n5tb2rqr), including raw material testing data, data about hardware repairs for both robots (including back-up robot instance data), and reliability data breakdowns for each hardware set. Further information including video documentation is available on request.

## Figures:

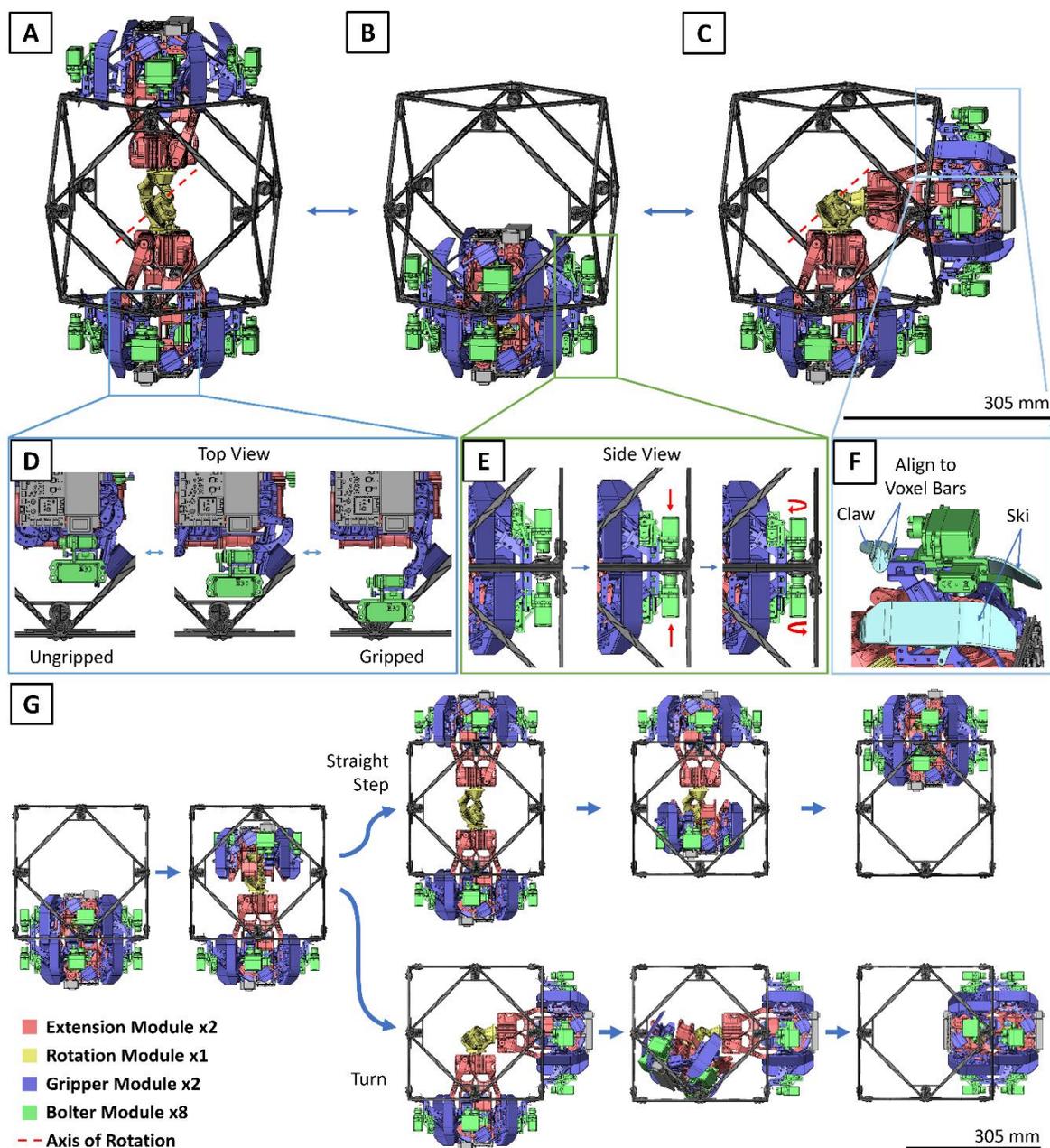


**Fig. 1. Overview of the structural system.** (A) Individual injection-molded faces have alignment features for inter-voxel connections and robotic gripping (B). Six faces are assembled into the cuboctahedron unit (C). These units are connected face-to-face (D) using four captive and reversible genderless fasteners (E, F). The bounding box of resulting voxel building block is a cube with 304.8mm side lengths.

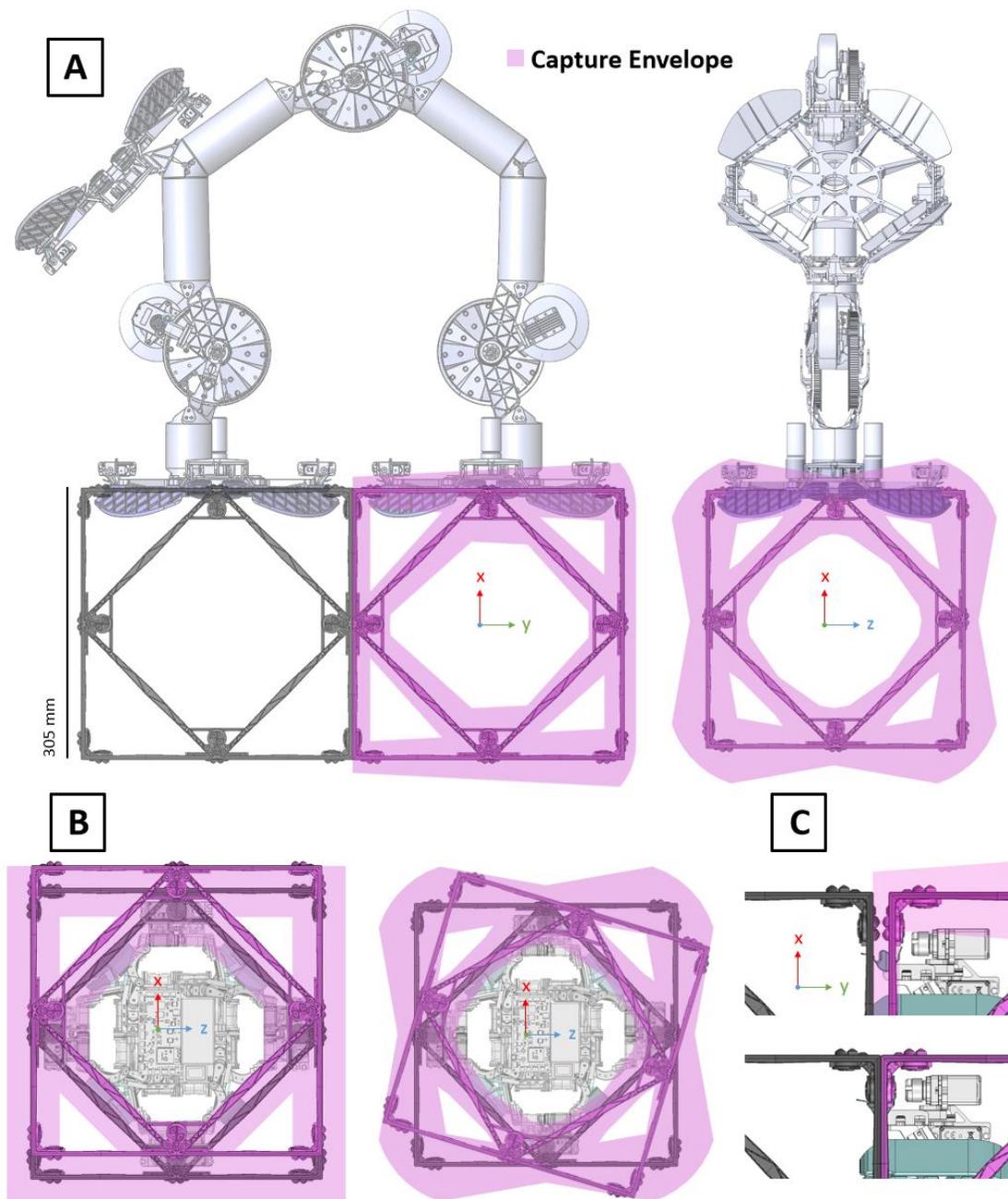


**Fig. 2. Overview of the external transport robot.** (A) The robot has three main drives and two yaw drives for locomotion; three gripper modules, one for cargo and two for the feet, allow the robot to carry voxels and grip to the structure, respectively. The foot modules have four petals and four grippers each to help with alignment during voxel gripping. (B) Operating as a team, two robots perform a voxel handoff, where the ‘crane’ robot (left) removes a voxel from the ‘cargo’ robot (right). The crane robot can then place the

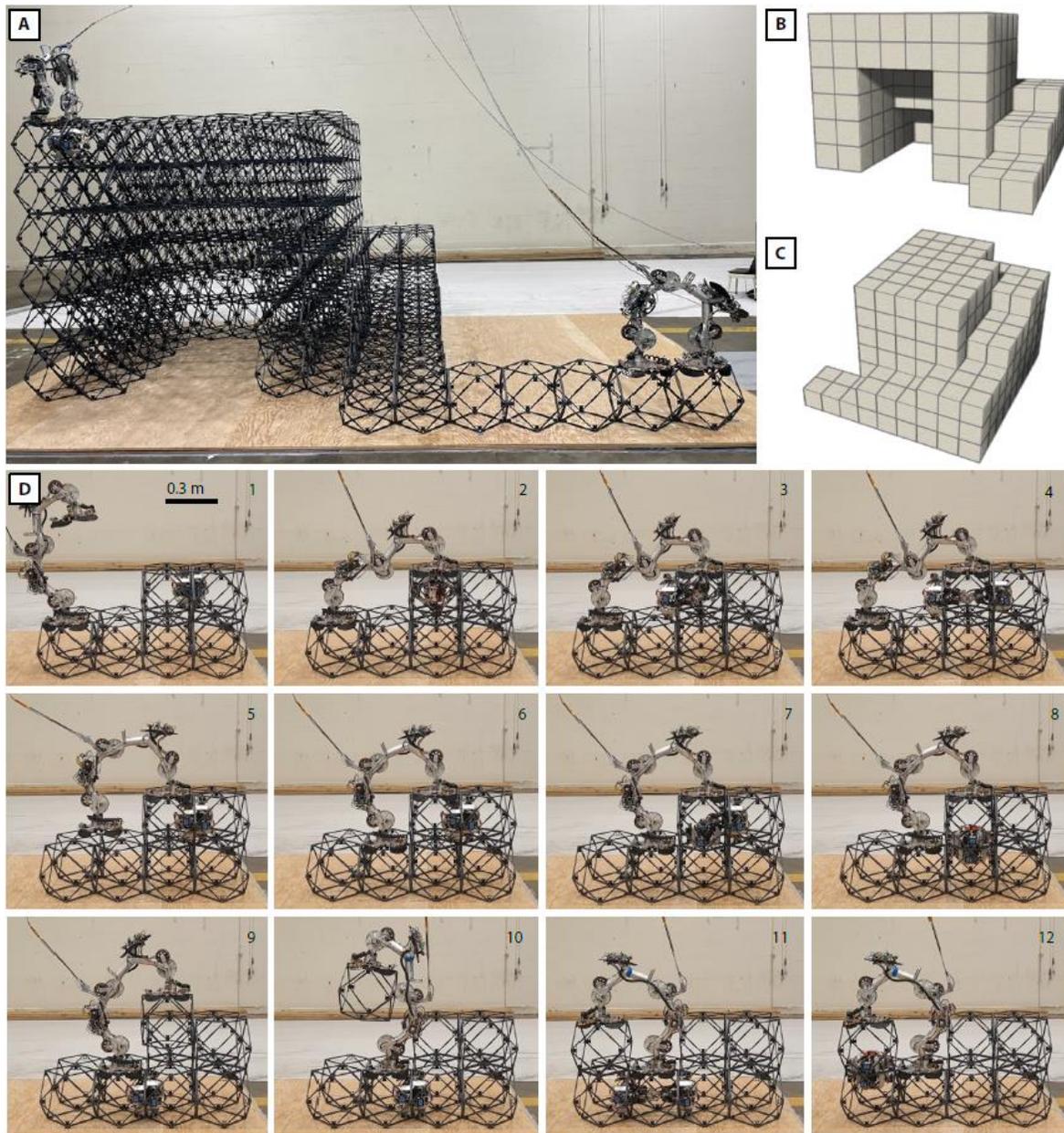
voxel (C). Both robots traverse the structure in a bipedal inchworm fashion (D).



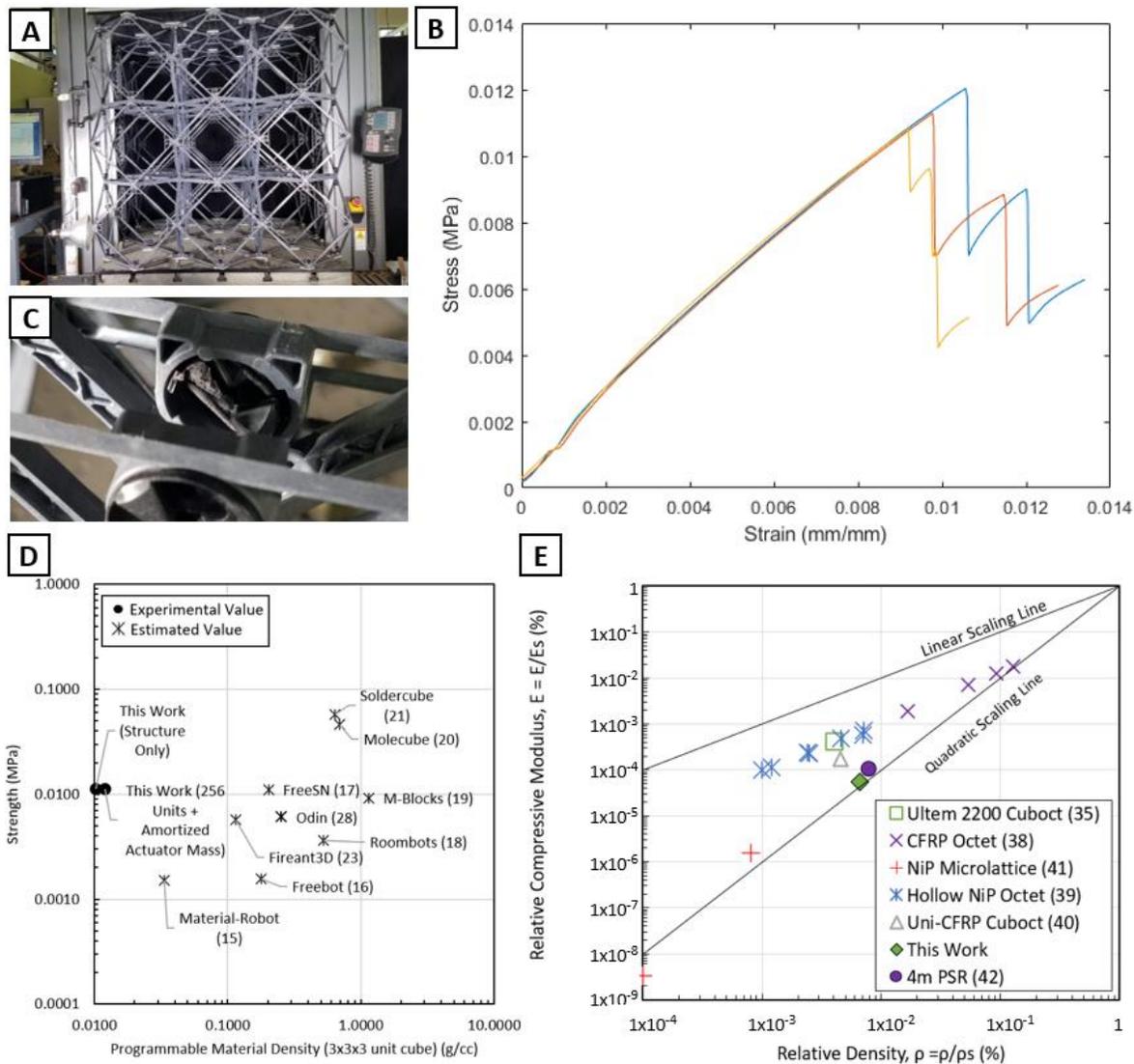
**Fig. 3. Overview of internal fastening robot.** The robot uses two extension modules and a rotation module to extend (A), contract (B), and turn (C) within the structure. Gripper modules (D) allow the robot to grip adjacent faces of the lattice, and bolter modules (E) allow the robot to actuate the captive fasteners and join adjacent voxels. Alignment skis and claws are integrated onto the gripper modules (F) for locomotion and fine positioning of a newly placed voxel. To locomote between cuboctahedron faces through the lattice structure (G), the robot combines a series of extensions, contractions, and turns.



**Fig. 4. Overview of Alignment Requirements (Capture Envelopes).** The robots utilize alignment guides that help place and align the newly placed voxels for fastening. These capture envelopes occur in stages throughout the voxel placement procedure and represent the misalignment the guides can correct and accommodate. **(A)** shows the capture envelope for voxel placement. As the fastening robot locomotes into the voxel face represented in **(B)**, its skis accommodate both translational and rotational offsets within the voxel-to-voxel plane ( $X$  and  $Z$ ). The claws on the fastening robot grip the voxel for final alignment in **(C)** and enforce  $<0.2\text{mm}$  of misalignment (in  $Y$ ).



**Fig. 5. 256 build experiment and reconfiguration experiment.** (A) Completed build of 256 voxels shelter structure. This example structure highlights the ability to build overhangs. Front isometric view (B) and back isometric view (C) of a CAD model of the finished structure. (D) This time-sequence of images shows transport and fastening robots separating a unit cell from the existing structure (1-9), relocating it to a new position (9-11), and then rejoining it to the structure (12). Commands were sent wirelessly via a centralized controller. Tether to external robot provided only power (no communication). Movie 1 shows the full reconfiguration.



**Fig. 6. Lattice Material Testing and Performance.** (A) Three different 3x3x3 voxel assembly samples were tested in compression. (B) Stress-strain curves showed consistent behavior. (C) Failure occurred at the inter-voxel fastener. This photo shows the broken fastener tooth in the separated voxel-voxel interface. (D) Comparison of the strength with other self-reconfigurable material systems. (E) Comparison of the stiffness with other architected materials and estimates of state-of-the-art assembled space truss structures.

**Tables:**

**Table 1. Summary of Robot Faults for the 256-Voxel Build.** Each robot fault was categorized by the module that caused the fault. Avionics and tether interference fault data is shown here, but not included in the command success rate calculations.

	<b>Type of Fault</b>	<b>Total Instances</b>	<b>No Touch Recovery</b>	<b>Touch Recovery</b>	
<b>Cargo SOLL-E</b>	gripper	147	100	47	
	locomotion	18	0	18	
	belt skip	53	0	53	
	avionics	21	12	9	
	tether interference (operator error)	12	1	11	
	All	218	100	118	
	<b>Total Number of Motions</b>		<b>14185</b>		
	<b>Command Success Rate</b>		<b>98.5%</b>		
	<b>Success Rate including No Touch Recovery</b>		<b>99.2%</b>		
	<b>Crane SOLL-E</b>	gripper	4	1	3
locomotion		6	2	4	
belt skip		0	0	0	
avionics		7	6	1	
tether interference (operator error)		1	0	1	
All		10	3	7	
<b>Total Number of Motions</b>		<b>478</b>			
<b>Command Success Rate</b>		<b>97.9%</b>			
<b>Success Rate including No Touch Recovery</b>		<b>98.5%</b>			
<b>MMIC-I</b>		gripper	27	12	15
	extension and rotation modules	19	4	15	
	avionics	12	4	8	
	All	46	16	30	
	<b>Number of Steps Taken</b>		<b>711</b>		
	<b>Command Success Rate</b>		<b>93.5%</b>		
	<b>Success Rate including No Touch Recovery</b>		<b>95.8%</b>		
		bolter	33	26	7

	<b>Number of Faces Joined</b>	<b>596</b>
	<b>Command Success Rate</b>	<b>94.5%</b>
	<b>Success Rate including No Touch Recovery</b>	<b>98.7%</b>

**Movies:**

Movie 1: System overview video