MANUFACTURING

Ultralight, strong, and self-reprogrammable mechanical metamaterials

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

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 a demonstration of disassembly and reconfiguration. The assembled structural lattice material exhibits ultralight mass density (0.0103 grams per cubic centimeter) with high strength and stiffness for its weight (11.38 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 the potential for self-reconfiguring autonomous metamaterials for diverse applications.

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. Because they can be reused 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 macroscale three-dimensional (3D) 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 toward this vision of macroscale 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. Toward the spirit of general macroscale 3D reconfigurable materials, we distinguish programmable material examples that are fully 3D, “self-programmable,” and “reprogrammable.” For material generality, we restrict consideration to systems that yield a 3D material that has non-zero compression and tensile strength (placement or reconfiguration of unbonded bricks is not considered because it cannot 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 frames 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 to progress on fully synthetic 3D self-reprogrammability (*7*, *14*), but open challenges remain in the 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 reconfiguring (*28*), have faced similar implementation challenges, likely because of the challenges of joint design and interface alignment.

---

[1]NASA Ames Research Center, Moffett Field, CA, USA. [2]KBR, Moffett Field, CA, USA.
*Corresponding author. Email: kenny@nasa.gov (K.C.C.); christine.e.gregg@nasa.gov (C.E.G.)
†Present address: Stanford University, Stanford, CA, USA.

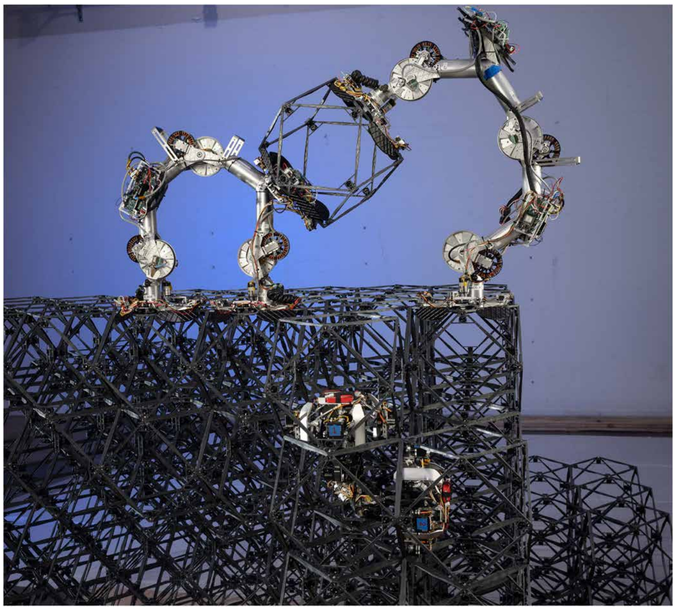
Our self-reprogrammable system combines mechanical metamaterial building blocks and two types of robots (*29*, *30*) to create a reconfigurable structural-robotic system (fig. S1). We separate the actuation and structural components to achieve system mass efficiency on par with static structural materials (*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 a 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 used. Alignment features on the voxels and robots ensure that each aligns correctly with the 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 the 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 here. The following sections describe the system in detail.

System components

The cuboctahedron structural unit cell voxels (Fig. 1C) were constructed from six injection-molded chopped carbon fiber–reinforced polymer (StattechNN-40CF) square faces (Fig. 1A and fig. S2) (corner-to-corner dimension of 304.8 mm, yielding a voxel with a bounding box edge dimension of 304.8 mm). Material and Methods figs. S3 and S4 describe the characterization of the constituent fiber-reinforced injection-molded material (data in tables S5 and S6). These square faces were preassembled into voxels using conventional bolts and integrated alignment features (Fig. 1B) located at the corners of the faces. Voxels connected face to face (Fig. 1D) using four custom injection-molded genderless reversible fasteners (Fig. 1F) (*34*) that were held captive in the voxel face (Fig. 1E). O-rings between the fastener and voxel face housing ensured that the preinstalled 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 (±8.71 N) tension load (fig. S5 and table S1). They were designed to apply a preload to the inter-voxel joint that increased the joint strength beyond the fastener load (like any preloaded bolted joint).



Movie 1. System overview video.

Two external transport robots (Fig. 2) collaborated to deliver voxels from a supply depot (voxel source) to the build front and to 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. 2A). To align to voxels, the robot used 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, whereas the other foot accessed nearby voxel faces on 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. 2D). The crane robot could unload a voxel from the cargo gripper (Fig. 2B) and place or remove a voxel at any available lattice position (Fig. 2C). 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 (Fig. 3 and fig. S6) (*29*) climbed through the structure and actuated captive fasteners between voxels (Fig. 3E). In the contracted position (Fig. 3B), the robot occupied a square face of the cuboctahedral lattice. By extending (Fig. 3, A and C) and gripping adjacent faces (Fig. 3D), the robot moved between faces and traversed in three dimensions (Fig. 3G). Dual-purpose alignment features on the robot body (skis) and grippers (claws) (Fig. 3F) ensured that 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. Whereas 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 Materials and Methods.

The robots had multistage 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 because 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

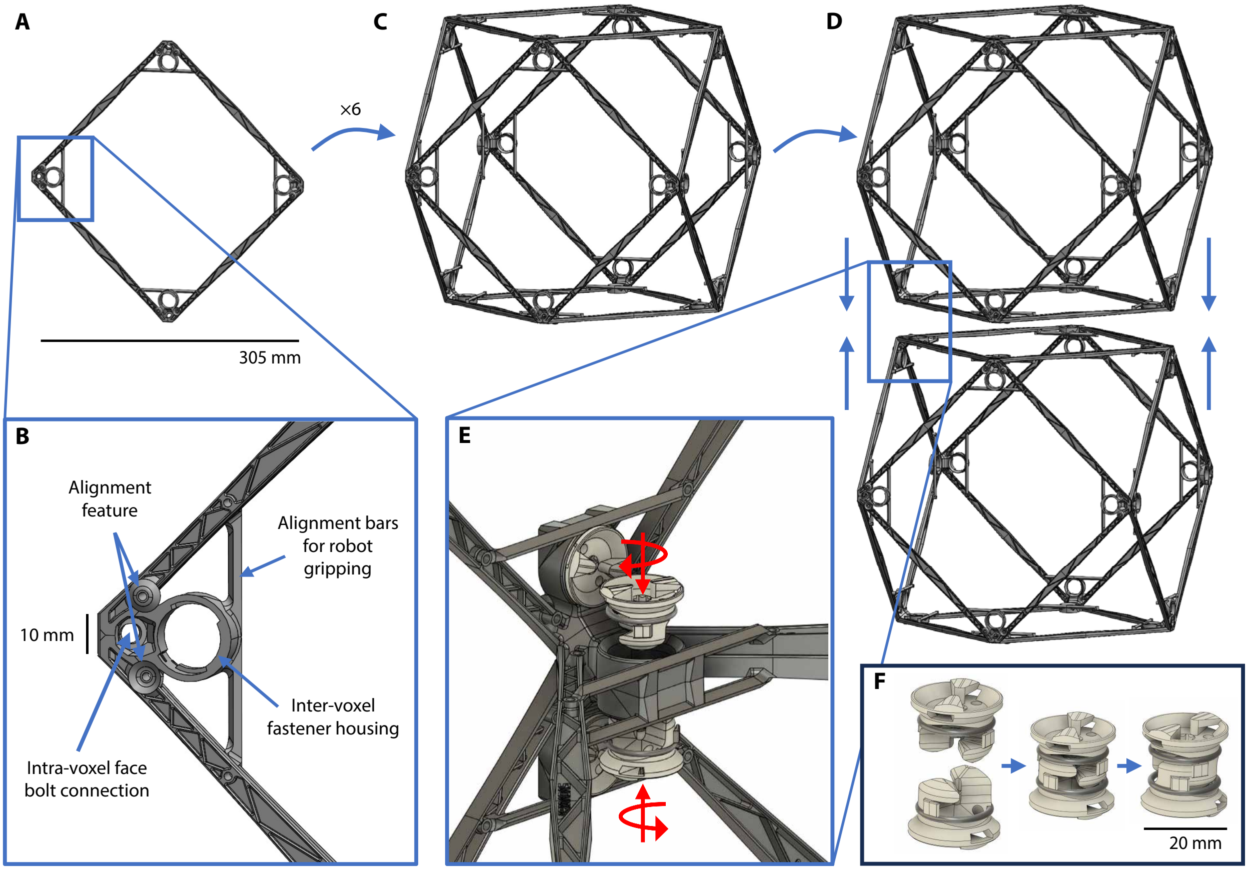


Fig. 1. Overview of the structural system. (**A**) Individual injection-molded faces have alignment features for inter-voxel connections and robotic gripping (**B**). Six faces are assembled into the cuboctahedron unit (**C**). These units are connected face to face (**D**) using four captive and reversible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

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 of 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° from the ideal position (Fig. 4A). 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 <0.1 mm (in x and z) (Fig. 4B). 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 <0.1 mm at full grip (Fig. 4C). 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 the 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 overconstrained 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 to 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 voxels 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 the final alignment and attached the new unit cell to the existing structure. On the basis of a manually generated build plan consisting of a 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

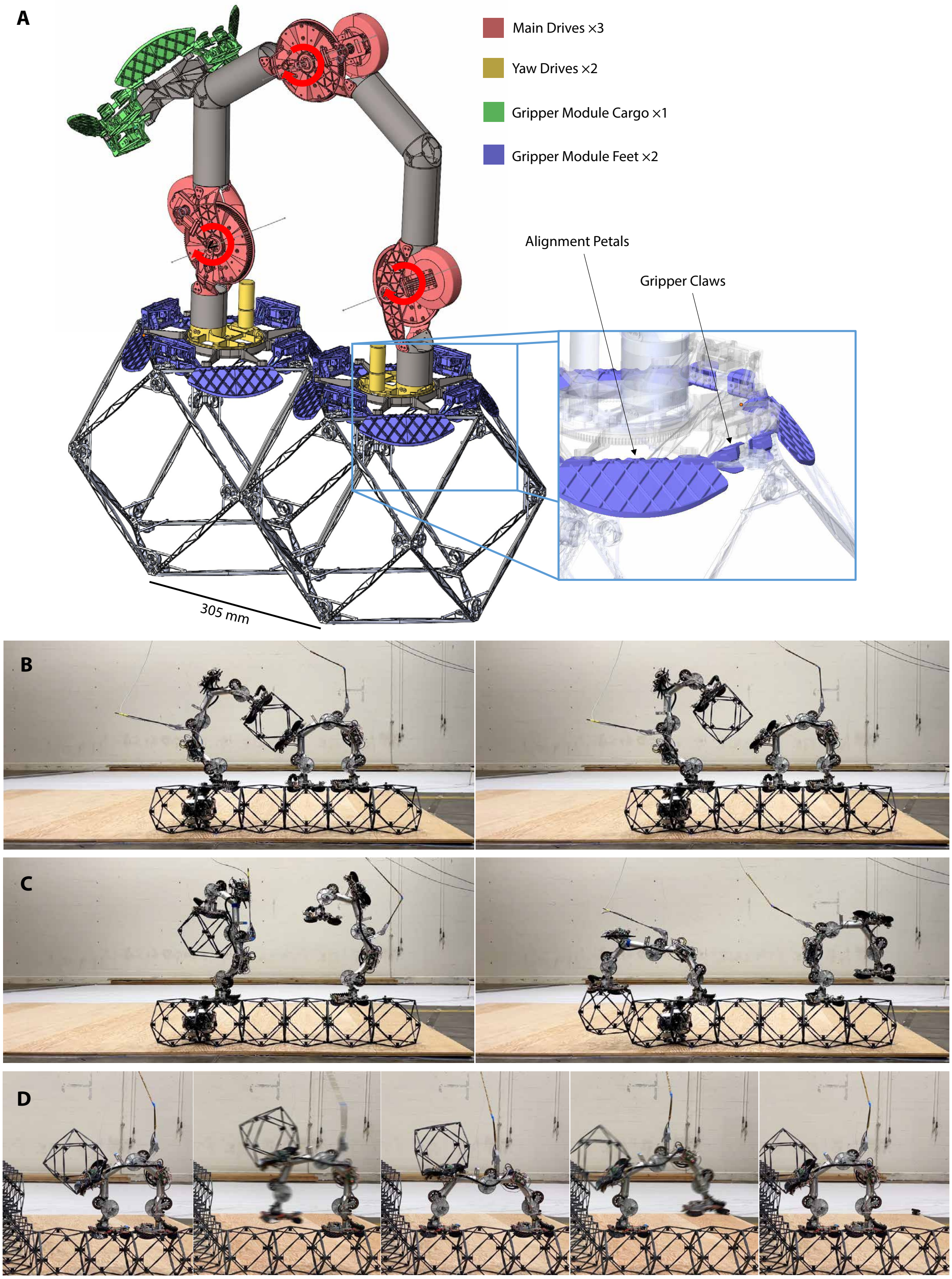


Fig. 2. Overview of the external transport robot. (**A**) The robot has three main drives and two yaw drives for locomotion; three gripper modules, one for cargo and two for the feet, allow the robot to carry voxels and grip the structure, respectively. The foot modules have four petals and four grippers 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**).

if a fault was triggered). The control system issued primitive commands representing discrete actions, such as “step forward,” “step up,” or “turn right 90°.” The robots then executed preestablished 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

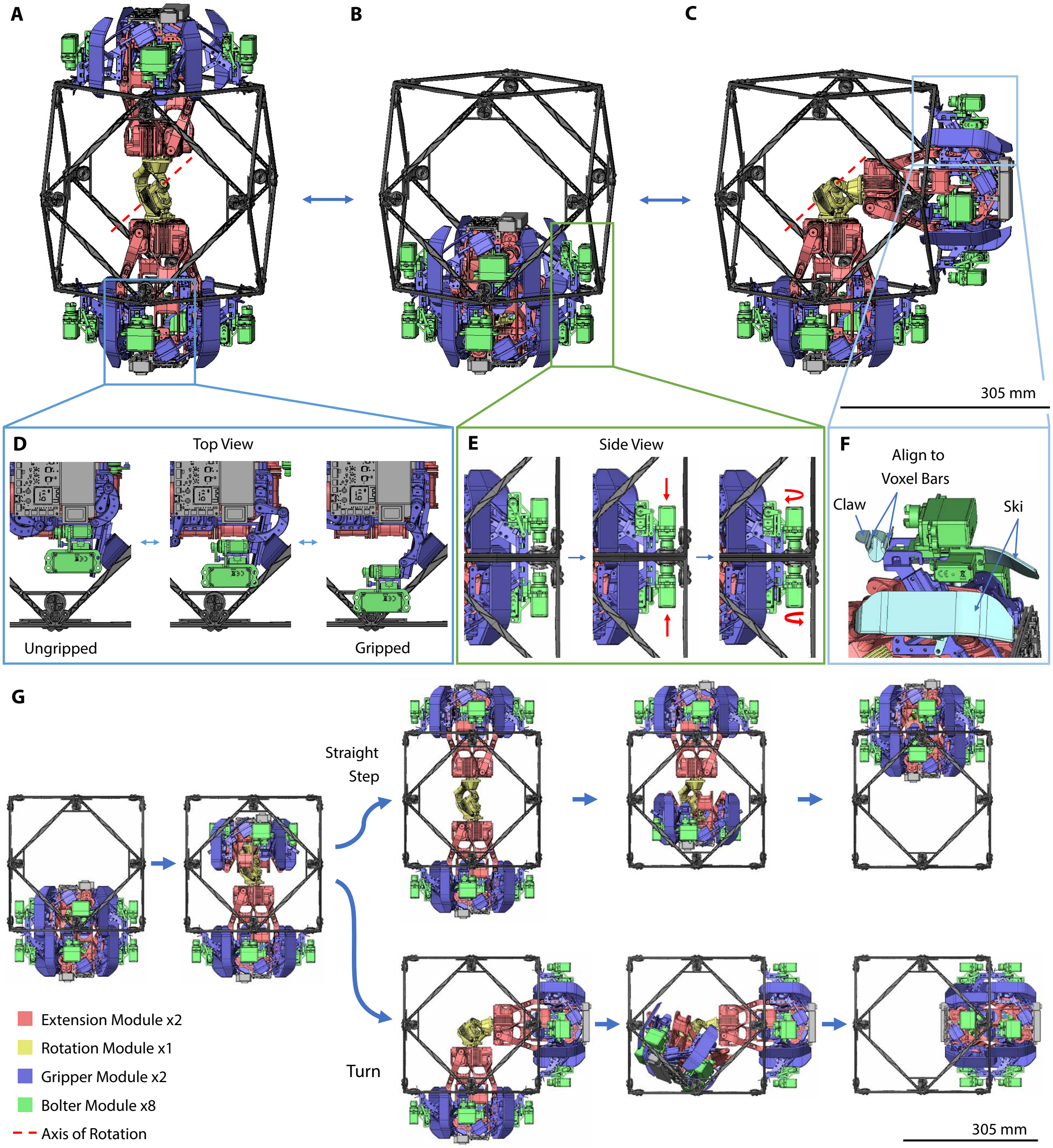


Fig. 3. Overview of the internal fastening robot. The robot uses two extension modules and a rotation module to extend (**A**), contract (**B**), and turn (**C**) within the structure. Gripper modules (**D**) allow the robot to grip adjacent faces of the lattice, and bolter modules (**E**) allow the robot to actuate the captive fasteners and join adjacent voxels. Alignment skis and claws are integrated into 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.

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, a contracted position limit switch, an inertial measurement unit (IMU), and servo electrical current monitoring. The system did not use 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 an assembly throughput rate of 20,000 mm^3/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

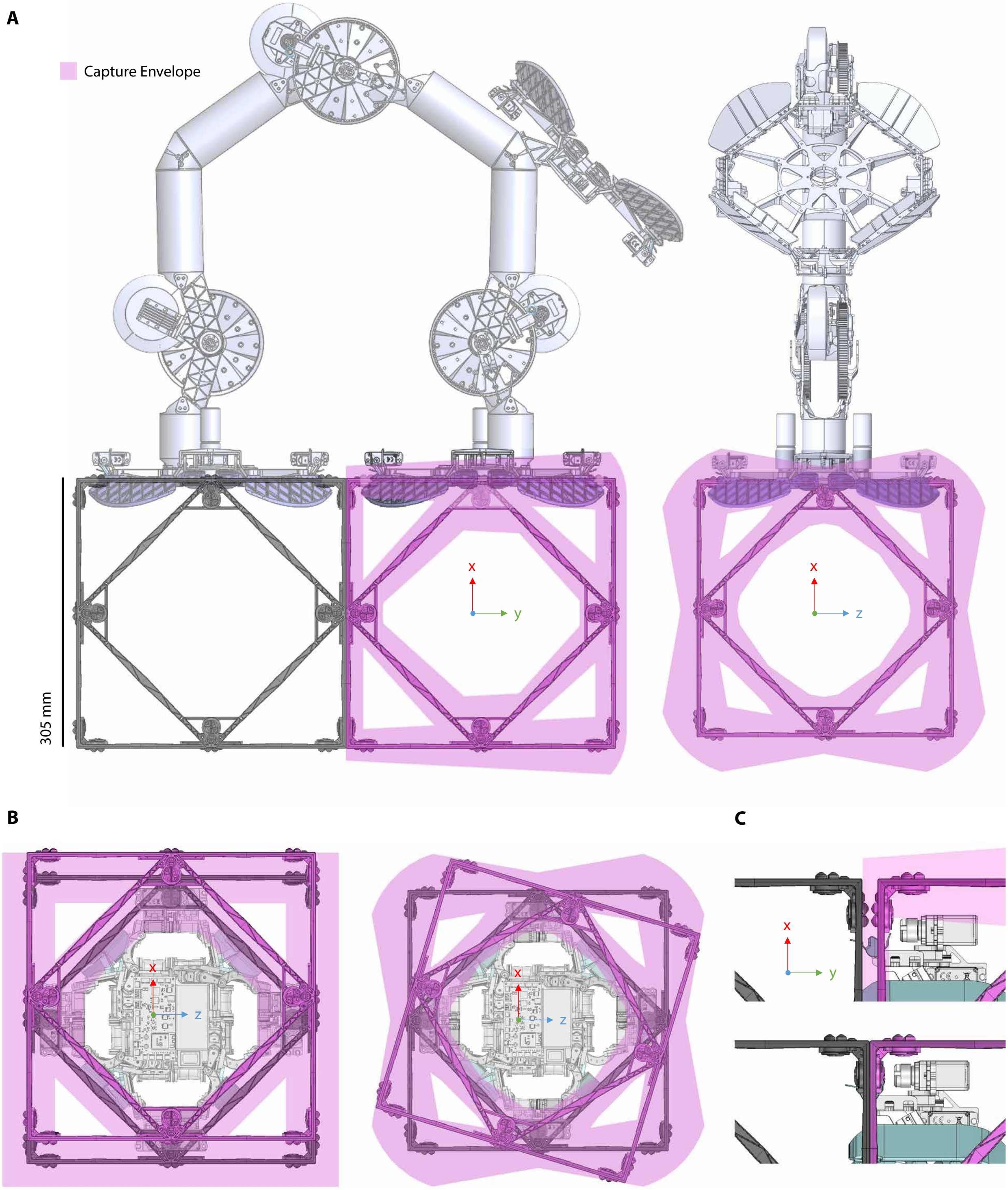


Fig. 4. Overview of alignment requirements (capture envelopes). The robots use alignment guides that help place and align the newly placed voxels for fastening. These capture envelopes occur in stages throughout the voxel placement procedure and represent the misalignment the guides can correct and accommodate. (**A**) The capture envelope for voxel placement is shown. 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 mm of misalignment (in *y*).

voxels, and bolting fasteners. Distance traveled varied by robot role because the fastening and the crane transport robots primarily remained at the build front (Table 2). The cargo transport robot traveled 4624 body lengths (3.15 km), 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 (CG), calculated for each movement primitive using the robotic mass distributions (tables S3 and S4).

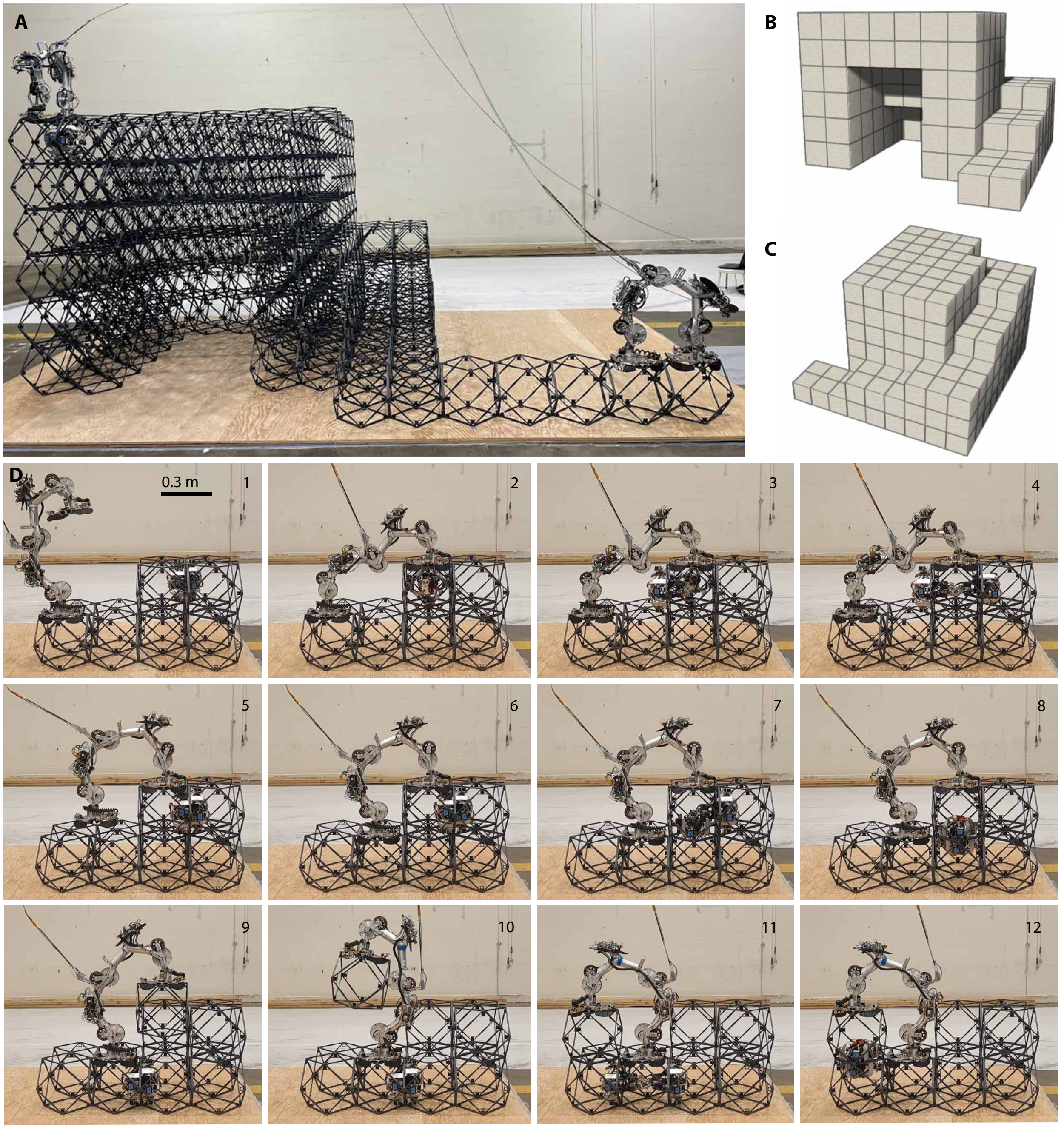


Fig. 5. The 256-voxel build experiment and reconfiguration experiment. (**A**) Completed build of 256-voxel shelter structure. This example structure highlights the ability to build overhangs. Front isometric view (**B**) and back isometric view (**C**) of a digital model of the finished structure. (**D**)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

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 include either 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 timeout, main joint target timeout (when the motor did not reach its target within the 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),

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 are shown here but not included in the command success rate calculations. For the internal robot, locomotion and bolter faults are separated.

Fault category	Type of fault	Total instances	No touch recovery	Touch recovery
		Cargo SOLL-E		
Locomotion	Gripper	147	100	47
	Locomotion	18	0	18
	Belt skip	53	0	53
	Avionics	21	12	9
	Tether interference (operator error)	12	1	11
	Total	218	100	118
		Crane SOLL-E		
Locomotion	Gripper	4	1	3
	Locomotion	6	2	4
	Belt skip	0	0	0
	Avionics	7	6	1
	Tether interference (operator error)	1	0	1
	Total	10	3	7
		MMIC-I		
Locomotion	Gripper	27	12	15
	Extension and rotation modules	19	4	15
	Avionics	12	4	8
	Total	46	16	30
Bolting	Bolter	33	26	7

faults were divided into two categories: those due to mechanical issues with the robots and those due to communication loss timeout or tether interference that was unsuccessfully managed by the operators, requiring a pause in operations. Because systems in an operational environment would use 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 because of fatigue from a small radius feature and belt skipping because of 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 because of the failed support strut or the location of the gripper sensors, which led 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 (reorientations 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 a 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 (the system paused and notified the operator, followed by the 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. 5D 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 not only to the reprogrammability of the structure and adaptability but also to the scalability and the 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.

Table 2. Command success rate calculations. Command success rate calculations did not include avionics and tether interference fault data. Command success rates of the internal robot are separated into locomotion and bolting.

	Cargo SOLL-E	Crane SOLL-E	MMIC-I	
			Locomotion	Bolting
Total number of motions	14,185	478	n/a	n/a
Number of steps taken	n/a	n/a	711	n/a
Number of faces joined	n/a	n/a	n/a	596
Command success rate	98.5%	97.9%	93.5%	94.5%
Success rate including no touch recovery	99.2%	98.5%	95.8%	98.7%

During early development testing of the robots, robots demonstrated the ability to manipulate broken voxels (because of 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 the 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*) and multilabeled 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 corresponds 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 corresponds 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, and 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 the feasibility of a solution, we used a multilabeled 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 that 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 was initiated from a base layer of voxels. The robots started from their designated base positions, built the tower, and returned to their bases afterward. 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 use 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 three voxel–by–three voxel–by–three voxel (Fig. 6A) cubes were tested in compression with a mechanical testing machine to evaluate strength and stiffness (Fig. 6B). Failure of the structure occurred at the fastener (Fig. 6C) before buckling of the struts, suggesting that design iterations to strengthen the inter-voxel connection could further increase strength. The resulting structure had a strength of 11.38 kPa (SD ±0.51 kPa), corresponding to an average failure load of more than 9000 N. This exceeded estimated strengths for many previously demonstrated reconfigurable programmable matter systems at orders of magnitude less density (0.0103 to 0.0121 g/cm^3, depending on the amortization of the robot mass). Figure 6D 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 and fig. S8). The structural system had a stiffness of 1.1129 MPa (SD ±0.779 kPa), which at ultralight density achieves a relative stiffness regime typical of high-performance carbon fiber lattice truss systems for space applications. Because no similar reconfigurable robotic systems reported

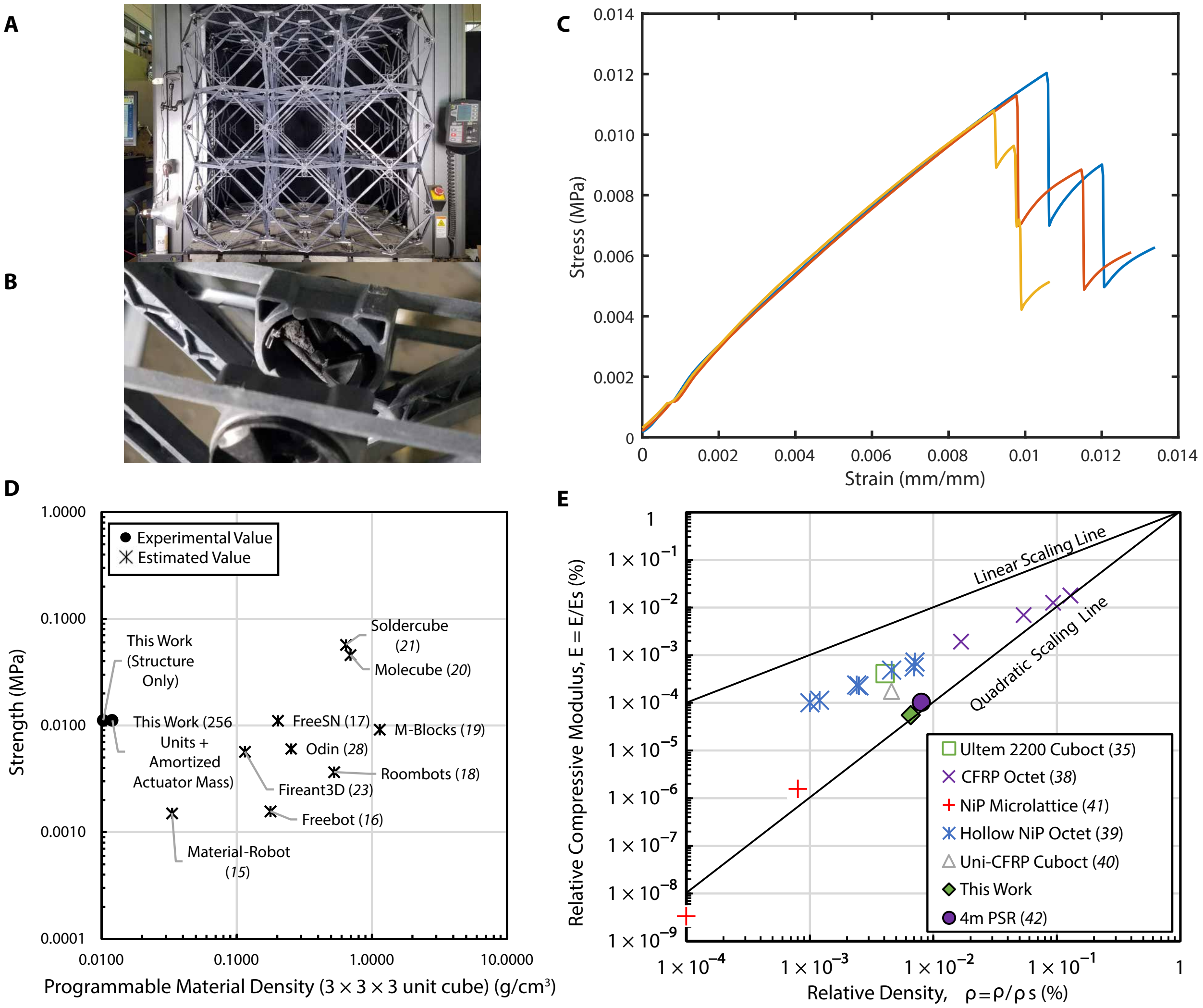


Fig. 6. Lattice material testing and performance. (**A**) Three different three voxel–by–three voxel–by–three voxel assembly samples were tested in compression. (**B**) The failure occurred at the inter-voxel fastener. This photo shows the broken fastener tooth in the separated voxel-voxel interface. (**C**) Stress-strain curves showed consistent behavior. (**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.

assembly stiffness for comparison, Fig. 6E compares the relative compressive modulus of this work with those of 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 ultralight density, ρ. For many applications, including those typical for space structures, minimum mass performance indices do not scale directly with specific stiffness (E/ρ) but rather with $E^{1/2}/\rho$ or $E^{1/3}/\rho$ (*43*). This is the basis of the efficiency of trusses and lattice materials. By exceeding the ideal quadratic performance (Fig. 6E), 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.* (*45*) 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. Using the 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 4-m-diameter truss aperture (*42*) made from high-performance carbon fiber struts with a mass of 85.7 kg, a depth of 0.5 m, 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/cm^3. Normalizing with the reported strut stiffness of 16.9×10^6 psi (116.5 GPa) yielded a relative density of 1.02×10^{-4}. Further examples of this rationale and simulations showing the accuracy of this analytical method are available in (*46*). 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. On the basis of architected lattice theory, these results suggest that at ultralight 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 the 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 to further extend the structural performance of the system. Previous work on assembled architected lattices shows that increased performance can be reached by using higher-performance constituent materials like continuous carbon fiber (*38*, *40*, *43*, *47*). In addition, although the chosen voxel geometry and cuboctahedron lattice have a favorable combination of stiffness and features that make voxels simpler to assemble (including generous clearances for robotic end effectors) (*44*, *48*), we do not claim that they are 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 and 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, the ease of adding robots to the system allows for increased reliability because 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 that these can be mitigated with design refinement. Future implementations of this type of system may still benefit from the integration of additional sensing to further increase reliability, to 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 with 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. Because 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 used in industry. Challenges of conventional approaches [managing global and local reference frames, alignment with computer vision, metrology, and 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 of robotic, sensing, and computational complexity needed to reliably meet 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^3/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*). Although 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 allow any target geometry to be built and provide 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, reuse, 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 ultralight, high-performance structural regime and solving the "big systems" problem (*7*) enable new applications in infrastructure, aerospace, and exploration. The effect of revolutionizing engineered material life cycles 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. Because the injection-molded faces have a knit line at the center of each strut, the test coupons were injection-molded such that there were specimens both with knit lines (type I and type II) and without knit lines (type III) (fig. S2). Type III data were used to normalize the lattice performance for relative stiffness metrics because 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 are 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 (extension rate, 12 mm/min). Table S1 shows the break load of each fastener. Because 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 preloaded joints).

The square injection molded faces were assembled into voxels using 10 to 32 stainless steel bolts and nuts at the corners of the faces. Three voxel–by–three voxel–by–three voxel assembly specimens were hand-assembled using manually actuated fasteners. The specimens were cycled between 0 and 1500 N compressive force at 36 mm/min and 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 could be observed where the mass of the compressive plate (980 N) was taken up and a spherical seat connecting this plate to the load cell was properly loaded. All curves were zeroed in relation to this preload, and modulus measurements were taken after this preload (which represents even loading of the lattice). Failure occurred at the root of the fastener tooth, which was predicted by finite element analysis of the fastener. All failures were on the center voxel column. Ultimate strength was calculated from the maximum load. A chord modulus was reported during the consistent loading phase of the stress-strain curve, which is conservative compared with the auto-modulus algorithm previously described (table S7). Because previous experiments showed 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 shaping universal programmable matter are described in the literature, including for the lattice geometry used in this study (*52*). These capabilities are a subset of the capabilities of the robots used in this study. We used 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 the 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 a timing belt for gearing. Controls are handled with three custom onboard controller boards incorporating simple system-on-chip microcontrollers (ESP32 modules), two of which are paired with modular commercial off-the-shelf BLDC controllers (ODrive V3.6, one of which is used 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 Gear Motor). 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's 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 the 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 step entirely over a voxel while avoiding collisions between the robot and the 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 were 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 were pruned for manual and algorithmic planning purposes on the basis of the local configuration of the structure and other robots. SOLL-E's number and type of allowed motions are dictated by the needs of the assembly. Trajectories, and their waypoints, of the required motions, such as "step forward" or "step up", were hardcoded in the robot firmware and called by the central control station. Motion coordination among the three control boards of the robot was achieved by moving onto the next waypoint only when all the boards had 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°) to fully lock the two faces together (0.9 N m running torque to 1.5 N m, recognized as hard stop).

The robot is symmetric to allow it to move forward and backward (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 ready for bolting 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 and 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, and IMU) for system monitoring and fault checking.

The 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 the 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 (the build sequence paused when a low battery was indicated, the battery was manually swapped, and the build sequence resumed). A single MMIC-I battery set could operate the robot for approximately 3 hours, depending on the number of motions that were completed.

The time-lapse video in Movie 1 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 reattempt parts of the build after recovering from a fault or receiving maintenance (replacement of worn components, the need for which has been addressed in further robot versions). If a robot fault was triggered, then the robot autonomously performed a fault recovery routine and retried the command. If the robot was unable to complete self-recovery from a fault, then 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 took 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 was 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's body length. This metric normalizes the distance traveled for a robot by dividing the total distance traveled by the robot's 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 CG displacement for each primitive motion in the complete 256-voxel structure build plan. Each primitive has a start and end configuration, and the distance between the CG for each configuration gives the displacement. These values were calculated using a digital model with approximate mass distributions that did not account for variations from voxel payloads. The mass breakdowns for each assembly used for distance traveled calculations are shown in tables S3 and S4.

Reconfigurable robot literature example strength estimates

Because no literature examples of structural programmable matter reported the strength of the resulting matter, estimations were made using the 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 toward 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. S8A). 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. Because 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 three voxel–by–three voxel–by–three voxel structure. In the case of reconfigurable struts 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. S8B). 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*), because the elements were also face-connected cuboctahedron, an assumption of fastener-limited cuboctahedron lattice scaling could be applied.

In this case, the separation force was estimated from the size of the magnets as 55 ± 5 N. Because 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. For a cuboctahedron lattice, the compressive strength is expected to be approximately half of the tensile strength (*35*).

Supplementary Materials

This PDF file includes:

Figs. S1 to S8
Tables S1 to S9

Other Supplementary Material for this manuscript includes the following:

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. U.S.A.* **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 (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. *IEEE Robot. Autom. Mag.* **14**, 43–52 (2007).
8. W. McCarthy, Programmable matter. *Nature* **407**, 569 (2000).
9. B. Florijn, C. Coulais, 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, 2014), p. 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*, vol. 11340 of Lecture Notes in Computer Science (Springer International Publishing, 2019).
14. S. S. R. Chennareddy, A. Agrawal, A. Karuppiah, Modular self-reconfigurable robotic systems: A survey on hardware architectures. *J. Robot.* **2017**, 5013532 (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, 2020), 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, 2022), 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. *Rob. Auton. Syst.* **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, 2013), pp. 4288–4295.
20. V. Zykov, A. Chan, H. Lipson, Molecubes: An open-source modular robotics kit, in *IROS-2007 Self-Reconfigurable Robotics Workshop* (IEEE, 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, 2017), pp. 2717–2722.
23. P. Swissler, 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, 2020), 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, 2007), pp. 4171–4176.
25. D. Hjelle, H. Lipson, A robotically reconfigurable truss, in *2009 ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots* (ASME, 2009), pp. 73–78.
26. S. 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, 2009), 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, 2017), 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, 2008), 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, 2023), 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)* (IEEE, 2023), pp. 10736–10741.
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* (American Society of Mechanical Engineers, 2016).
32. B. Jenett, C. Cameron, F. Tourlomousis, 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, 2020), 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 in AIIDE'05: Proceedings of the First AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (2005), pp. 117–122.
37. F. Grenouilleau, W.-J. V. Hoeve, J. N. Hooker, A multi-label A* Algorithm for multi-agent pathfinding. *ICAPS* **29**, 181–185 (2019).
38. L. Dong, H. Wadley, Mechanical properties of carbon fiber composite octet-truss lattice structures. *Compos Sci Technol.* **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. Heard Jr., T. J. Collins, W. B. Fichter, R. E. Wallsomt, J. E. Phelps, Design and fabrication of an erectable truss for precision segmentedreflector application. *J. Spacecraft* **28**, 251–257 (1991).
43. S. M. Arnold, D. Cebon, M. Ashby, "Materials selection for aerospace systems" (Technical Memorandum 2012–217411, NASA, 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. *J. Spacec. 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, 2017), pp. 1–10.
47. M. F. Ashby, R. F. M. Medalist, The mechanical properties of cellular solids. *Metall. Mater. Trans. A* **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, 2019), 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, 2019), 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, 2020), pp. 8252–8259.

Acknowledgments

Funding: This work was supported by NASA Space Technology Mission Directorate, Game Changing Development Program (Automated Reconfigurable Mission Adaptive Digital Assembly Systems – ARMADAS – Project). **Author contributions:** Conceptualization: K.C.C., C.E.G., O.I.B.F., G.T.T., and I.W.P. Methodology: K.C.C., C.E.G., O.I.B.F., G.T.T., D.C., T.J.O., and I.W.P. Material and structural characterization: K.C.C., O.I.B.F., and C.E.G. Robotic development and demonstration: K.C.C., C.E.G., O.I.B.F., G.T.T., D.C., T.J.O., I.W.P., and M.E.O. Planning, scheduling, simulation, and algorithms: K.C.C., O.I.B.F., and C.E.G. Funding acquisition: K.C.C. and E.M.T. Project administration: C.E.G., K.C.C., and E.M.T. Supervision: K.C.C. and E.M.T. Writing—original draft: C.E.G. and K.C.C. Writing—original draft (supplementary): C.E.G., O.I.B.F., D.C., and G.T.T. Writing—review and editing: C.E.G., O.I.B.F., G.T.T., K.C.C., D.C., M.E.O., I.K., and F.M.S. **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 (K.C.C., C.E.G., O.I.B.F., G.T.T., D.C., T.J.O., I.W.P., M.E.O., and E.M.T.), as well as the design and manufacturing of modular structural components (K.C.C. and C.E.G.). **Data and materials availability:** All data necessary to reproduce these results are available in the main text or the Supplementary Materials. Raw data files are available on Dryad (DOI: 10.5061/dryad.3n5tb2rqr). Further information including video documentation is available on request.

Submitted 14 April 2023
Accepted 18 December 2023
Published 17 January 2024
10.1126/scirobotics.adi2746

Science Robotics

Ultralight, strong, and self-reprogrammable mechanical metamaterials

Christine E. Gregg, Damiana Catanoso, Olivia Irene B. Formoso, Irina Kostitsyna, Megan E. Ochalek, Taiwo J. Olatunde, In Won Park, Frank M. Sebastianelli, Elizabeth M. Taylor, Greenfield T. Trinh, and Kenneth C. Cheung

Sci. Robot. **9** (86), eadi2746. DOI: 10.1126/scirobotics.adi2746

Science Robotics (ISSN 2470-9476) is published by the American Association for the Advancement of Science. 1200 New York Avenue NW, Washington, DC 20005. The title *Science Robotics* is a registered trademark of AAAS.