PRINTED & MUSCULOSKELETAL ROBOTICS

Robot metabolism: Toward machines that can grow by consuming other machines

Philippe Martin Wyder[1]*, Riyaan Bakhda[2], Meiqi Zhao[2], Quinn A. Booth[3], Matthew E. Modi[3], Andrew Song[1], Simon Kang[1], Jiahao Wu[1], Priya Patel[1], Robert T. Kasumi[1], David Yi[1], Nihar Niraj Garg[1], Pranav Jhunjhunwala[1], Siddharth Bhutoria[2], Evan H. Tong[2], Yuhang Hu[1], Judah Goldfeder[2], Omer Mustel[2], Donghan Kim[2], Hod Lipson[1]

Biological lifeforms can heal, grow, adapt, and reproduce, which are abilities essential for sustained survival and development. In contrast, robots today are primarily monolithic machines with limited ability to self-repair, physically develop, or incorporate material from their environments. While robot minds rapidly evolve new behaviors through artificial intelligence, their bodies remain closed systems, unable to systematically integrate material to grow or heal. We argue that open-ended physical adaptation is only possible when robots are designed using a small repertoire of simple modules. This allows machines to mechanically adapt by consuming parts from other machines or their surroundings and shed broken components. We demonstrate this principle on a truss modular robot platform. We show how robots can grow bigger, faster, and more capable by consuming materials from their environment and other robots. We suggest that machine metabolic processes like those demonstrated here will be an essential part of any sustained future robot ecology.

INTRODUCTION

Biological organisms operate as open systems: They absorb material from their environment and expel waste (1). This process is the basis for the long-term resilience of biological organisms over their lifetime (2, 3). Progress in artificial intelligence has advanced robots' ability to adapt by learning new behaviors, but has left the robots' physical morphology fixed and monolithic. Typical robots today cannot increase in size and complexity, adapt, or self-repair. In contrast, biological lifeforms developed the ability for physical adaptation, repair, and replication, including absorbing and expelling material, long before any form of intelligence ever emerged (4–6). In light of that, artificial intelligence, although important, may just be one piece of the puzzle of true robot autonomy: robot self-sufficiency. For robots to become resilient and sustainable in the long term, we must develop processes that allow them to act as open systems and develop physically by consuming, expelling, and reusing material from their environment. We call this process robot metabolism.

Unlike traditional robot manufacturing processes, where robots may be involved in the process of making robots in a variety of ways, a robotic adaptation process qualifies as robot metabolism if it satisfies two criteria: First, robot metabolism cannot rely on active physical support from any external system to accomplish its growth; the robot must grow using only its own abilities. The only external assistance allowed is that which comes from other robots made of the same components. Second, the only external provision to robot metabolism is energy and material in the form of robots or robot parts. No new types of external components can be provided. In the case of the platform used in this work, material comes in the form of robot modules and energy in the form of electricity stored in each module's batteries.

The concept of robot metabolism raises more questions than we can answer here. Thus, we focused on a set of key challenges: self-assembly, self-improvement, recombination after separation, and robot-to-robot assisted reconfiguration. In this work, we demonstrate the potential of this approach and introduce a robot platform capable of achieving it. We believe that this is the first demonstration of a robot system that can grow from single parts into a full three-dimensional (3D) robot, while systematically improving its own capability in the process and without requiring external machinery.

The choice of robotic building blocks is key as it spans the ultimate space for all possible designs. Biological lifeforms comprise only about 20 amino acids assembled into polypeptides during protein synthesis, ultimately giving rise to innumerable proteins and millions of self-sustaining lifeforms (7). Similarly, modular robots constructed from a finite set of simple, standardized components give rise to diverse functional structures and adaptive mechanisms. We believe that imitating nature's methods, rather than merely its results, will lead more fundamental innovation in robotics. Replicating animals and humans in the form of robot dogs and humanoid robots is ultimately limiting. Thus, the robot building blocks need to be designed with the capacity for robot metabolism. Once developed, platforms capable of robot metabolism provide a physical counterpart to self-improving artificial intelligence. Thus, we open the possibility of robots changing their own form to ultimately overcome the limitations of human ingenuity.

We introduce the Truss Link, a robot building block designed to enable robot metabolism. The Truss Link is a simple, expandable, and contractible, bar-shaped robot module with two freeform magnetic connectors on each end. Animating any structure, Truss Links form robotic "organisms" that can grow by integrating material from their environment or from other robots (see Fig. 1). We show how two substructures can combine to form a larger robot, how 2D structures can fold into 3D shapes, how robot parts can be shed and then be replaced by another found part, and how one robot can help another "grow" through assisted reconfiguration.

[1]Mechanical Engineering, Columbia University, 220 S. W. Mudd Building, 500 West 120th Street, New York, NY 10027, USA. [2]Computer Science, Columbia University, 450 S. W. Mudd Building, 500 West 120th Street, New York, NY 10027, USA. [3]Electrical Engineering, Columbia University, 1310G S. W. Mudd Building, 500 West 120th Street, New York, NY 10027, USA.
*Corresponding author. Email: philippe.wyder@columbia.edu

Fig. 1. Robot metabolism allows machines to "grow." Robot modules can grow by consuming and reusing parts from their environment and other robots. This ability, essential to biological lifeforms, is crucial for developing a self-sustaining robot ecology. This paper demonstrates the above developmental sequence in detail: from individual modules to a fully assembled ratchet tetrahedron robot.

This work includes some results previously shared at the IEEE Re-Mar 2024 conference (*8*). In our conference paper, we shared a limited, hardware-focused treatment of the Robot Link, i.e., the Truss Link without its free-form attachment/detachment mechanism. In this work, we present our Truss Link capable of robot metabolism, including its orientation agnostic, passively actuated, permanent magnet attachment/detachment mechanism (see Fig. 2), our simulation and corresponding quantitative results, the ratchet tetrahedron formation as the final stage of our multistage robot development experiment, the demonstrated improvement at every stage across all developmental transformations as shown in movie S2, the shedding of dead links, and the assisted tetrahedron formation.

Truss Link

Truss Links can be used to build modular robots. Modular robot systems comprise multiple parts called modules, links, or cells that can self-assemble or be assembled to achieve an objective. The Truss Link is the basic building block of our modular robot system. Toshio Fukuda sparked a new generation of research, when he introduced modular robotics in 1988 (*9*). Modular robots promise increased versatility, configurability, scalability, resiliency, and ability to self-reconfigure and evolve (*10–12*). In addition, robot modularity could make robots cheaper if the modules were mass-produced (*10*). Modular robots are potentially resilient as a result of their redundancy and modularity, rather than mere material strength.

Modular robots can be classified as self-reconfiguring or manually reconfigurable robots (*13*). Self-reconfiguring robots can attach and detach from other modules automatically, while manually reconfigurable robots must be assembled by an operator. Truss Links enable modular self-configuring robots. A single Truss Link is capable only of motion in one dimension and, therefore, is limited to crawling forward and backward. Once a multilink topology such as a triangle or tetrahedron has been formed, the system becomes fully controllable in 2D or 3D, respectively.

As truss robots, Truss Links form "scaffold-type" structures and have expanding and contracting prismatic joints (see Fig. 2, A and B) rather than rotational ones as they are found in popular cubic-shaped models (*11*). Spherical and cubic robot models have the drawback of forming dense structures, making assembling large

robots difficult. Recent developments in modular robotics have shown increased interest in both truss-style and free-form modular robots. Spinos *et al.* and Park *et al.* (*14–16*) introduced the first truss robot capable of self-reconfiguration. Prior truss modular robots such as Morpho and Odin were limited by their attachment mechanism from self-reconfiguring (*17–19*). Both of these systems required connector cubes to join modules.

Many of the well-known cubical modular robot designs, such as Molecubes, M-Blocks, and Smores-EP, had power-sharing or communication channels built into their connectors and, as a result, were limited to a discrete set of attachment angles (*20–23*). Free-form modular robots such as the spherical FreeBot and FreeSN changed this by excluding electronic contacts from their connector; instead, they used a simple magnetic connector with infinite attachment possibilities (*24*, *25*). We chose a free-form style connector design to allow Truss Links to effectively self-assemble (see Fig. 2, C to G).

By combining free-form connectors with a truss-style module design, we created a self-assembling platform that forms sparse lattices rather than dense structures. The Truss Link's free-form magnetic connector allows it to attach freely from a wide range of angles without requiring precise alignment. The self-aligning magnet sphere allows multiple connectors to attach to each other, as shown in Fig. 2 (E to G). In our experiments, we successfully operated topologies that had up to four connector connections, and we manually assembled topologies with up to six connector connections. The mass of a Truss Link robot scales linearly with the number of Truss Links, while the pull-away force between connectors scales at a lower rate. Thus, 3D structures with more than four connectors connected at a point are more prone to failure. Despite these limitations, the Truss Link is the first truss-style modular robot capable of self-assembly and self-reconfiguration.

RESULTS

Our results demonstrate that it is possible to form machines that can grow physically and become more capable within their lifetime by consuming and recycling material from their immediate surroundings and other machines. While these results are still nascent, they suggest a step toward a future where robots can grow, self-repair, and
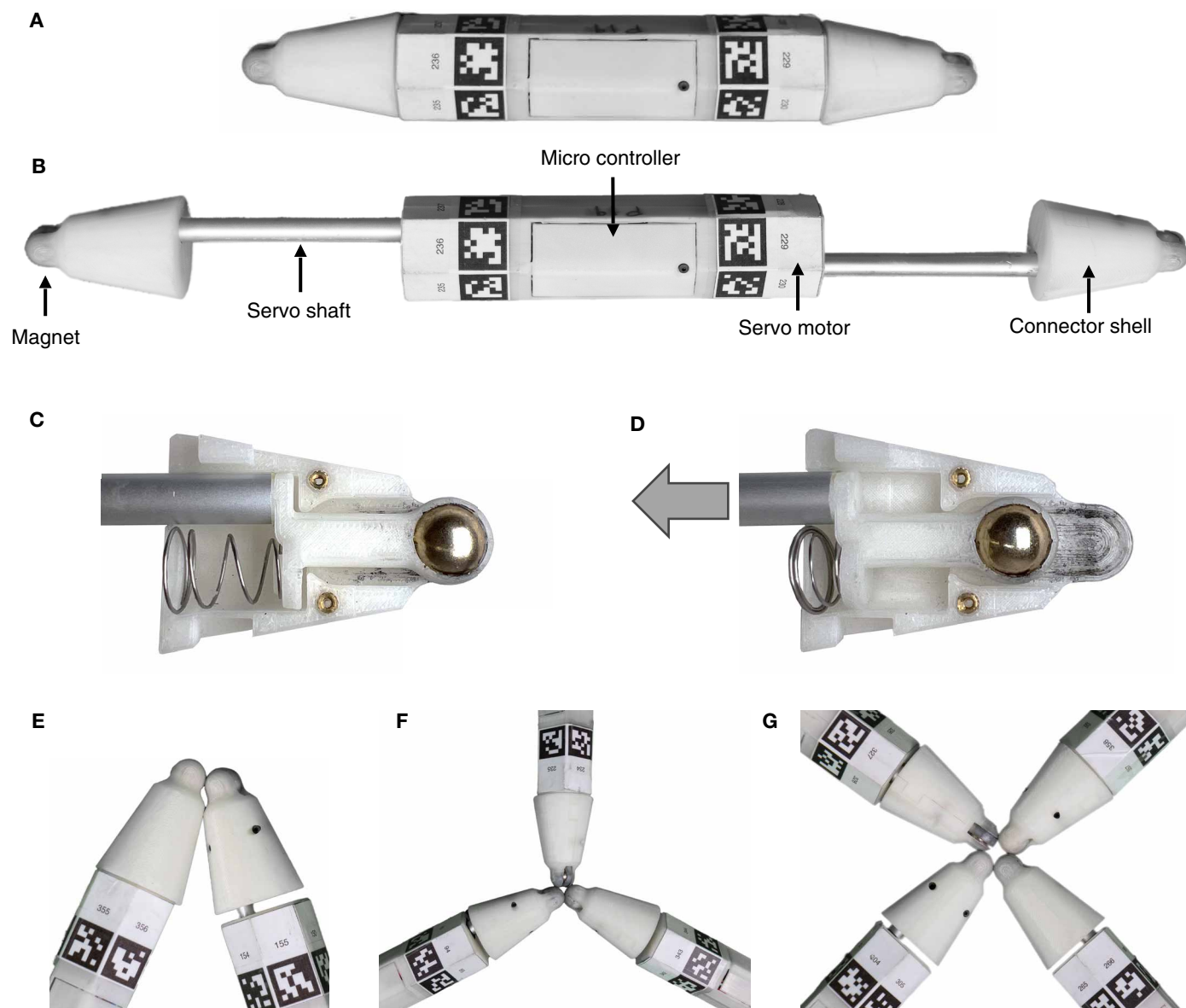
Fig. 2. Truss Links can expand and contract, attach and detach, and connect to multiple other Truss Links at once. (**A**) A contracted Truss Link is 28 cm long and weighs 280 g (**B**). When fully expanded, a Truss Link can increase its length by over 53% to 43 cm. Images (**C** and **D**) show the interior of the magnet connector in an active state with the magnet exposed at the tip and a fully-contracted, i.e., nonactive state with the magnet retracted, respectively. The conical compression spring inside the connector resets the magnet connector to the active state after retracting it, so the Truss Link is ready to connect again. The spherical neodymium magnet is held in position by a magnet holder. The magnet holder allows the magnet to rotate freely to rotate to an equilibrium position when approached by another magnet. This mechanism ensures a strong connection between multiple links from a wide and continuous range of angles. We show connections between (**E**) two, (**F**) three, and (**G**) four connectors.

adapt instead of being purpose-built with the vain hope of anticipating all use cases. Robot platforms capable of robot metabolism open the door to the development of machines that can simulate their own physical development to achieve an objective and then execute that physical development. By acting as open systems, robots capable of robot metabolism bear the potential of forming self-sustaining robot ecologies that can grow, adapt, and sustain themselves, given a continued supply of robot material.

The Truss Link is the first modular truss robot capable of robot metabolism. To start, we demonstrate the Truss Link's capacity for self-assembly from individual parts, forming a three-pointed star

and a triangle, and by integrating existing substructures, forming a diamond-with-tail from a triangle and a three-pointed star. Second, we quantify the probability of random topology formation in simulation given similar randomized initial conditions used in our physical demonstration. Third, we show how Truss Link structures can recover their morphology after separation due to impact via self-reconfiguration or self-reassembly. Fourth, we introduce a way for a ratchet tetrahedron morphology to shed a "dead" Truss Link and replace it by picking up and integrating a found link. Last, we expand beyond the individual robot and demonstrate how a ratchet tetrahedron robot can assist a 2D arrangement of links to form a tetrahedron.

The Truss Links were operator controlled in all physical Truss Link experiments using a custom keyboard interface. The interface allows the operator to send commands to selected Truss Links or trigger preprogrammed open-loop control scripts. The preprogrammed controllers allowed us to topple tetrahedrons or make ratchet tetrahedrons and tetrahedrons crawl.

Multistage robot development

The multistage robot development experiment tested whether a 3D structure capable of absorbing and integrating more material could be formed from independent 1D robotic building blocks. If possible, this would lay the foundation for truss robots capable of growing in complexity due to self-assembly and physical development. Next, we quantified the probability of our robotic building blocks randomly assembling into the topologies shown in the multistage robot development experiment in simulation. These probabilities provide a reference for the likelihood of achieving these developmental transitions.

To test our hypothesis, we investigated which environmental conditions facilitated self-assembly. In nature, we see environmental factors crucial to successful development, with early-stage development being most sensitive to environmental conditions. Bird embryos require a hermetically sealed egg to grow, while mammals require a temperature-stabilized womb. Similarly, Truss Links' ability to develop and form new structures is influenced by environmental factors. Identifying a suitable environment was crucial for achieving robot development from basic parts.

In our simulation environment, we explored what type of world environment would allow us to transition from 2D robot structures to 3D robot structures, in particular the diamond-with-tail to tetrahedron transition. Through experimentation, we found that this transition is more likely to succeed if a diamond-with-tail crawls off a ledge (see Fig. 3 ledge between B-b and B-c), and has an obstacle to lean up against (see black vertical obstacle in Fig. 3, B-c) while folding in on itself, connecting the tail of the diamond-with-tail to its tip (see $t = 217$ to 231 s in Fig. 3D). Once we identified a suitable environment, we then built a four-stage (see Fig. 3B and fig. S1), 3.9-m-long and 0.9-m-wide experiment environment, mimicking the simulated environment. To enable the diamond-with-tail to tetrahedron transition, a ledge followed by an obstacle was placed between stages 3 and 4 (see Fig. 3, B-b and B-c).

The experiment involves a total of seven Truss Links. Six Truss Links start on the first stage (Fig. 3B-a), and the seventh Truss Link is waiting to be picked up by the tetrahedron on the 3rd stage (Fig. 3B-c). Throughout the experiment, there are five topological transitions. First, the formation of a triangle and a three-pointed star from six individual links, followed by the triangle absorbing the three-pointed star to form a diamond-with-tail (see Fig. 3B-a). Next, the diamond-with-tail forms by crawling off a ledge and folding in on itself (see Fig. 3B-b). Last, similar to the tetrahedral mechanism discovered by Lipson and Pollack in (26), the tetrahedron transitions into a tetrahedron ratchet configuration by picking up a found Truss Link and using it as a walking stick (see Fig. 3B-c).

Each transition in this experiment is designed to produce a more capable topology. Individual links can only crawl forward and backward in 1D space. A triangle and a three-pointed star can both navigate in 2D space and, therefore, can circumvent obstacles that a single Truss Link could not. In contrast with a triangle or a three-pointed star, a diamond-with-tail can overcome a 25-mm-tall ledge

and can fold itself into a tetrahedron. A tetrahedron can move in three dimensions by toppling onto obstacles that were inaccessible to previous topologies. A ratchet tetrahedron increases its walking speed by over 66.5% on a 10° slope compared to a tetrahedron (see Fig. 4) (see movies S2 and S3). After assessing its feasibility in simulation, we successfully reproduced every transition of the experiment on the physical platform. Our experiments demonstrated that three independent links can combine to form a triangle and a three-pointed star configuration. Next, we showed that a triangle can connect to and integrate a three-pointed star to form a diamond-with-tail shape that can further fold itself into a tetrahedron. Last, we demonstrate how a tetrahedron robot can consume a found Truss Link and integrate it into a tetrahedron-ratchet configuration (see Fig. 3E).

Simulated versus average real world crawling speeds

We compared the crawling performance between topologies: individual Truss Link, triangle, tetrahedron, and ratchet tetrahedron. We replicated the 10° slope environment and the gates used on the physical robot in our PyBullet simulation. We normalized the speeds of all topologies by their body length: 28 cm for the individual Truss Link and 24.5 cm for the triangle, tetrahedron, and ratchet tetrahedron topologies. Further, we multiplied their body length per second speed by the time it takes each topology to execute one crawling cycle, i.e., "take a step". The cycle time is 16 s for the ratchet tetrahedron and 36 s for all other topologies. The physical friction conditions of the Truss Links on the carpet cannot be accurately replicated in the PyBullet simulation environment. The simulated ramp acts as a hard untextured surface with a lateral friction coefficient of 0.89 and spinning and rolling friction coefficients of 0.02 and 0.003, respectively. Please see our code repository for the corresponding simulation scripts to replicate these simulations.

The simulation data were filtered using a z-height threshold to exclude periods when the topology had moved off the platform. Each data point corresponds to a crawl cycle. Locomotion speed was estimated by fitting a linear regression to the y position and timestep data over four consecutive cycles, advancing the window by two cycles at each step. The resulting speeds were averaged, and their SDs were computed, then normalized to body lengths per cycle.

The results from the simulated runs for each topology, as well as the average body length normalized speeds of our physical experiments and their corresponding SD, are shown in Table 1. We find that the crawling speeds between topologies vary less in simulation. Most notably, the simulated triangle crawled faster than expected, while the simulated tetrahedron crawled slightly slower.

We observed that the physical Truss Links showed more backsliding behavior, especially the triangle. On the consistent and smooth surface in the simulation environment, the crawling speed of the triangle was slightly faster than that of the tetrahedron, while in our physical experiments, it was substantially slower. The weight shifting performed by the tetrahedron by moving its upper three links does not result in the same benefit on the hard simulated surface as it did on the soft carpet where the edges of the connectors would sink in. The simulated links do not suffer from manufacturing errors: slightly rotated motor shafts and rough edges on the Truss Link body that could result in added friction. The real triangle's crawling behavior was more brittle than that of the tetrahedron: it was more likely to get stuck due to a loose servo shaft or to rotate due to differences in friction between the two front Truss Links.
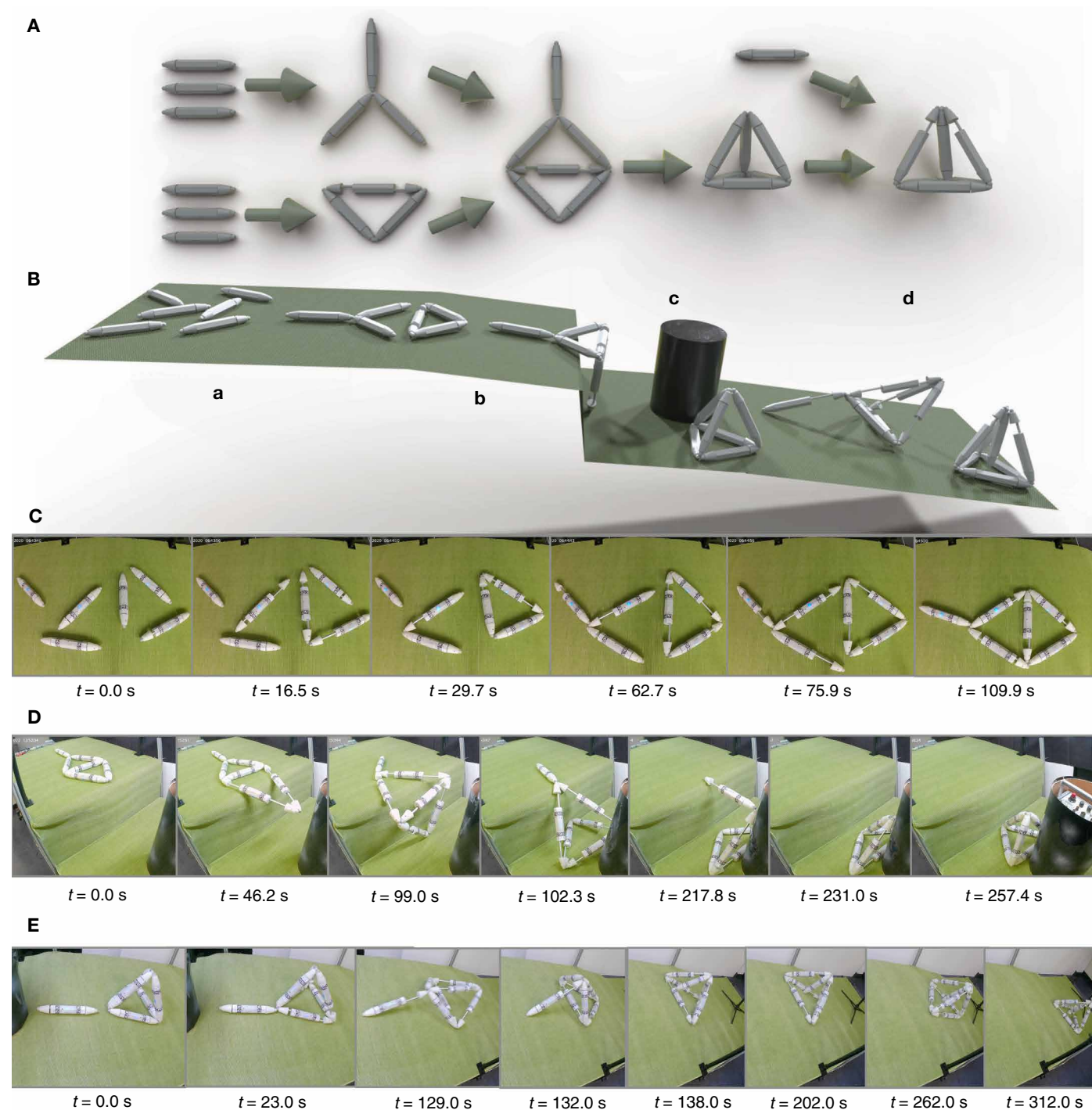
Fig. 3. Truss Links can develop 3D structures by absorbing and integrating material. (**A**) shows a series of topological transitions, starting on the left from a group of individual links and ending on the right with a ratchet-tetrahedron topology. Starting from six independent links, three links combine to form a three-pointed star shape, and the other three combine to form a triangle. Next, the triangle absorbs the three-pointed star by connecting to it and becomes a diamond-with-tail topology. The diamond-with-tail then folds itself into a tetrahedron. Next, the tetrahedron finds and integrates a free Truss Link by connecting and picking it up from the ground to form a ratchet tetrahedron. (**B**) shows the profile view of the experiment environment (not to scale), clarifying where each transition shown in (C) to (E) took place with section labels (B-a) through (B-d) as a reference. The frame sequences in (**C**), (**D**), and (**E**) show the formation of a diamond-with-tail, a tetrahedron robot, and a ratchet tetrahedron, respectively.

All topologies were faster in simulation than in the real world, which likely is a direct result of the difference in friction and surface texture. The ratchet tetrahedron benefitted most from the simulated friction condition and is faster than all other topologies on average, but highly volatile in its crawling speed. The ratchet tetrahedron in simulation and on the physical robot has a propensity to rotate along its vertical axis and deviate from its intended path. Given its instability, the ratchet tetrahedron crawling behavior does not lend itself well to open-loop operation. In contrast, the individual Truss Link and the tetrahedron demonstrated the most consistent crawling behavior in simulation with a smaller SD than the other two topologies.

Simulated morphology formation probabilities

Teleoperated, physical experiments do not highlight the difficulty of forming different morphologies. Thus, we quantified the formation probabilities of different morphologies in simulation. We simulated the morphological development experiment environment, spawned the Truss Links randomly in the same section of the experiment environment as the physical experiment, and randomized the control inputs. We added walls to the simulated experiment environment to prevent Truss Links from falling off. To track the morphologies during simulation, we hashed all magnets based on their x and y locations into a 2D occupancy grid with 16 mm-by-16 mm square cells and then considered all magnets that were within the same cell or within neighboring cells to be connected. On the basis of our empirical observation of the physical platform, this is a reasonable assumption since two magnets within that range would inevitably snap together. Next, we represented the morphology as a graph by
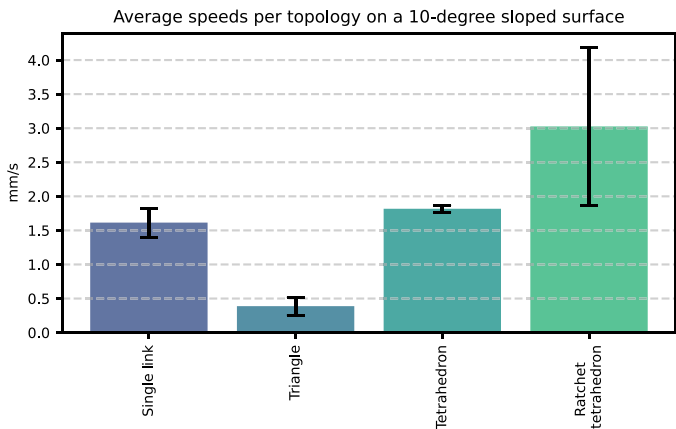
treating the links as edges and groups of connected connectors as nodes. Last, we computed the Weisfeiler-Lehman hash for each graph representing a specific morphology (see Supplementary Materials section on Morphological representation and tracking in simulation). The resulting formation probabilities provide a numerical reference for the likelihood of the transitions in our previous experiment occurring by chance without human assistance.

The analysis was conducted on 2000 random experiment runs, each limited to 20 min of simulation time. Experiments that were initialized with links already connected were excluded from the analysis and not counted toward the 2000 analyzed experiment runs. For each run, we stored the set of all morphologies that occurred during the simulation. From these data, we extracted the probabilities shown in Fig. 5.

The formation probabilities show that some but not all of the morphologies could be reproduced spontaneously from the random initial state with random motor commands within 2000 attempts. It becomes apparent that the formation of a diamond-with-tail is highly likely from the spawn locations chosen in the experiment, given that it occurred in 44.3% of the experiment runs. This high probability points toward an initialization bias, which was intentional since the initialization was supposed to mimic the one used in the physical experiment. However, it is worth noting that just 9.2% of the experiment runs exhibited a three-pointed star and a triangle simultaneously, indicating that most diamond-with-tail shapes were not formed as demonstrated on the physical robot by combining a triangle with a three-pointed star.

From the physical experiment, we learned that forming the tetrahedron and the ratchet tetrahedron is possible but challenging without an added controller. The tail link of the diamond-with-tail shape is only connected at one point and thereby position constrained but free to rotate around the connection point. Similar to an inverted pendulum, this was challenging for a human operator to learn. Over the course of 37 attempts, the environment was adjusted: for example, adjusting the distance and tilt of the cylindrical obstacle and flattening the carpet. At the same time, the human operator had to learn to control the robot using the keyboard interface (see fig. S4). After the last environment adjustment, the first successful tetrahedron was formed on the sixth attempt. Thus, we can conclude that more randomized runs and more simulation time would have produced a nonzero probability for the tetrahedron. Following this line of reasoning, Truss Links could "grow" on their own even if they acted randomly.

Damage recovery

Biological life's ability to self-heal by reforming broken bonds or growing back parts inspired us to attempt robot self-repair by reforming broken bonds between Truss Links. The magnetic connections between Truss Link connectors form predetermined breaking points, reducing the risk of physical damage to the Truss Link hardware

Fig. 4. Ratchet tetrahedron robots gain speed at the cost of consistency. The graph visualizes the locomotion speeds of a single Truss Link, a triangle, a tetrahedron, and a ratchet tetrahedron. The error bars show the SD from the mean. The experiment was conducted on a flat, carpeted, 10° decline.

Table 1. Simulated versus real single-direction locomotion speeds on a 10° downward slope in body lengths per cycle.

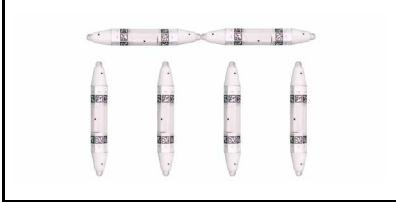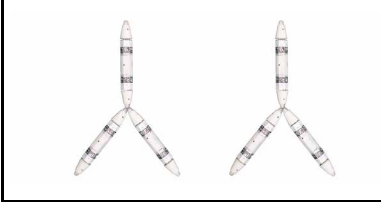|	Single link	Triangle	Tetrahedron	Ratchet tetrahedron
Simulated	0.3554 ± 0.0014	0.3382 ± 0.0446	0.3274 ± 0.0159	0.3867 ± 0.2397
Real	0.2070 ± 0.0275	0.0573 ± 0.0168	0.2674 ± 0.0063	0.1979 ± 0.1494
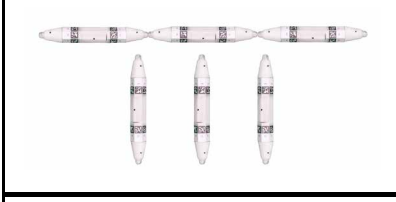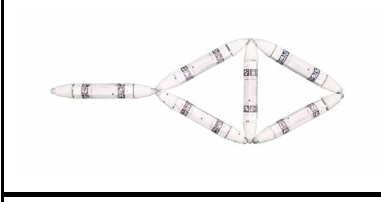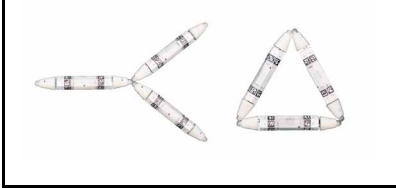
Formation	Probability	Formation	Probability
	100%		8.4%
	98.6%		64.35%
	97.6%		44.3%
	9.2%		0%

Fig. 5. Simulated random topology formation probabilities over 2000 20-min simulation runs.

from impact. In this section, we explore how this feature enables robots to recover their original topology after being separated upon impact. In our tests, we let triangle, three-pointed star, and diamond-with-tail robots crawl off the 30-cm-tall ledge between stages B-b and B-c of the experiment setup shown in Fig. 3, such that they disconnected on impact and then attempted to regain their original morphology.

For this experiment, we limited damage to a loss of the original topology due to broken connections between Truss Links. This is in contrast to the breaking or malfunctioning of Truss Links. We intentionally kept the drop height low to avoid damage to the Truss Links. In the case of a broken Truss Link the robot would need to get rid of the broken part and replace it with a functioning one, as shown in our next experiment.

The violent disconnections after impact and the slopes of the experiment environment resulted in hard-to-predict outcomes that were difficult to control for the operator. Thus, several reconstruction attempts were not successful. We share examples of successful shape recovery for all three topologies below.

The triangle is a fully constrained shape and, therefore, a naturally stable planar topology. As a result, the triangle resisted breaking any connections on several attempts. All triangle connections are strong two-connector connections without unconstrained degrees of freedom. If one connection did break, the other two would usually hold.

In one failed attempt, the triangle managed to break both connections with the back Truss Link, which, due to the sloped surface, rolled and re-connected at the connection point between the other two links, forming a three-pointed star. In another attempt, the back Truss Link broke a single connection, but the operator did not manage to re-form the triangle within the bounds of the filming setup and thus aborted the attempt.

A successful damage recovery sequence is shown in Fig. 6. Notice how, at $t = 84$ s, the triangle's back left connection gets disconnected due to the asymmetric fall. After extending its back Truss Link, the triangle recovers the connection by extending its front-right Truss Link.

In contrast to the triangle, the three-pointed star topology is under-constrained: All three links are only connected on one end. As a result, it is less predictable, harder to control, and more brittle. Several attempts failed spectacularly with links being flung down the ramp or rolling away, thereby making shape recovery impossible.

A successful sequence showing the damage recovery of a three-pointed star is shown in Fig. 7A. At $t = 80$s the three-pointed star drops and completely disconnects following the impact. The three-pointed star recovered its original form after the links rolled near each other, and the Truss Link facing in the 2 o'clock direction at $t = 82$ s shuffled itself to a 3 o'clock position ($t = 260$ s). The three-pointed star was able to crawl away after recovering its shape.
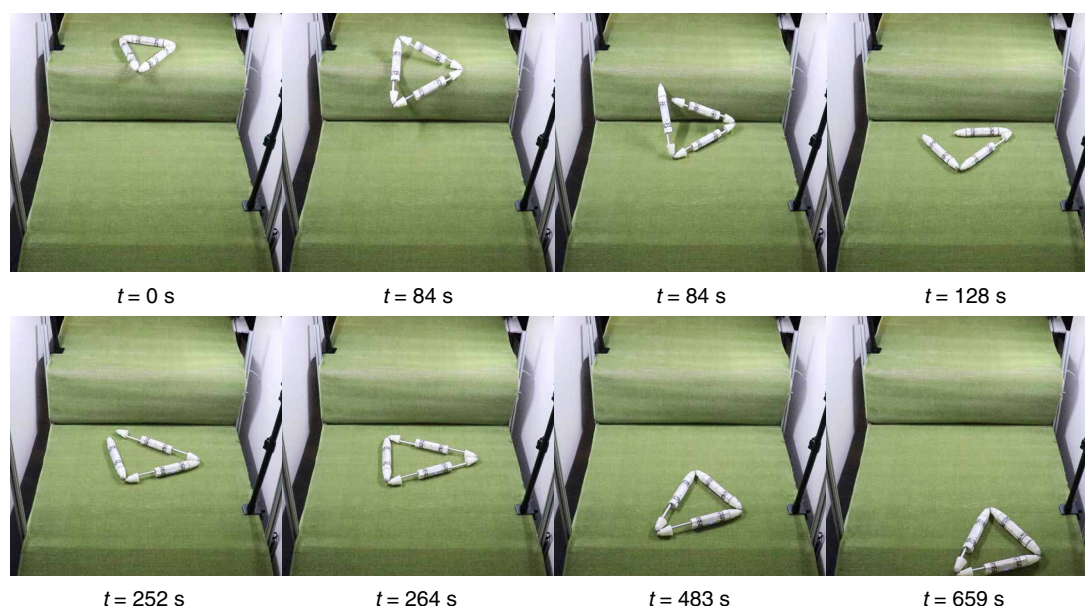
Fig. 6. A Truss Link triangle robot recovers its shape after impact. A Truss Link triangle robot crawls off a ledge, breaks a connection due to the impact, proceeds to recover its triangle shape, and crawls away.

To assess if a larger structure could recover its original shape, we conducted the experiment using the diamond-with-tail topology consisting of six links. Only one of the connections on the diamond-with-tail are two-connector connections; the other three connections are three-connector connections.

The diamond with tail structure is under-constrained and similarly unstable when falling as the three-pointed star. When falling off the ledge, the front of the structure crashes into the experiment surface, while the back end is still sliding or falling, adding additional force to the Truss Link connections and breaking them. As a result, the Truss Links further back in the structure can fall on top of the links in the front.

A successful recovery sequence of a diamond-with-tail that separated in the middle into a triangle and a three-pointed star is shown in Fig. 7B. The three-pointed star landed on top of the triangle and had to shuffle itself off of the triangle first before reconnecting. The three-pointed star managed to connect to the lower-right vertex of the triangle. After 4 min of maneuvering, the second Truss Link of the three-pointed star reconnected to the lower-left corner of the triangle. Last, the reformed diamond-with-tail moved itself off the ramp.

Replacing a "dead" Truss Link

Truss Link structures can self-assemble, but can they self-repair? In this experiment, we tested if a ratchet tetrahedron could recover from losing its ratchet Truss Link due to power loss. Truss Links are programmed to fully contract and detach by retracting the magnets inside the connectors once battery power drops below a critical threshold. Thus, similar to apoptosis in multicellular organisms (i.e., programmed cell death), the robot can shed a Truss Link that is no longer needed or threatens the robot's overall functionality.

In the frame sequence shown in Fig. 8, the ratchet tetrahedron first finds and connects to a replacement Truss Link with its right-front-bottom vertex. Next, as shown in Fig. 8, at $t = 36$ s and following, the ratchet Truss Link is triggered to execute its death sequence, where the Truss Link fully contracts both servos. This causes the ratchet Truss Link to let go of its connection and then roll away due to the environment slope at $t = 38$ s. Next, the tetrahedron topples first forward (see $t = 107$ s) and then to the right (see $t = 147$ s) to get into position to pick up the replacement Truss Link. Last, at $t = 192$ s, the tetrahedron picks up the new Truss Link, swings it inside itself ($t = 226$ s to 331 s), and then continues to use it as a ratchet at $t = 379$ s and following.

This experiment was conducted on stages three and four of the experiment (see Fig. 3, B-b and B-c). The experiment environment has a slope that is necessary to enable the tetrahedron to pick up the found Truss Link. The slope also has the benefit of allowing a shed Truss Link to potentially roll away and thereby not interfere with the process of picking up the replacement link.

Robot-to-robot assisted reconfiguration

Earlier, we have demonstrated how a ratchet-tetrahedron can be assembled from individual Truss Links. However, the transformation from diamond-with-tail to tetrahedron, as shown in Fig. 3, is not trivial and requires specific environmental conditions. Here, we study if, once a tetrahedron has been formed, the transition from a 2D flat pattern to a tetrahedron could be facilitated by robots assisting each other.

In this experiment, we identified a way to erect multiple consecutive flat patterns into tetrahedrons one after another, thereby significantly lowering the difficulty of forming more tetrahedrons after the first ratchet tetrahedron is formed. Inspired by the teardrop-shaped canyon cross sections found in Leprechaun Canyon, the experiment environment features a raised platform with a narrow opening and a sloped surface below. From this elevated position, the ratchet tetrahedron can assist other links to extend into the third dimension.

A frame sequence of this experiment, including multiple camera angles, is shown in Fig. 9. A ratchet tetrahedron can position itself
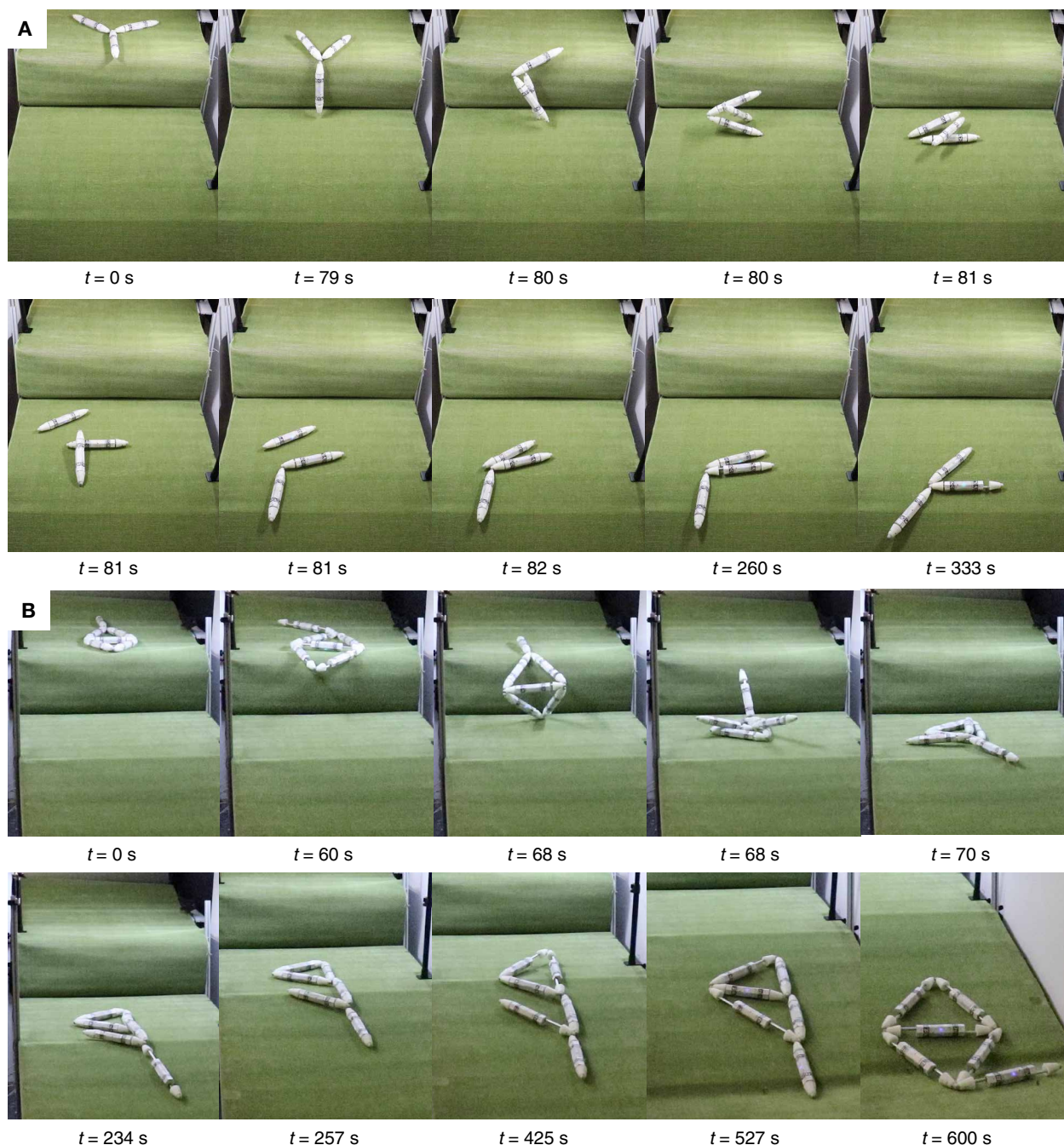
Fig. 7. A Truss Link–based three-pointed star and diamond-with-tail robot recover their original form after breaking connections due to impact. (**A**) A three-pointed star robot crawls off a ledge and breaks all Truss Link connections. The robot then regains a three-pointed star shape and crawls away. (**B**) A Truss Link diamond-with-tail robot crawls off a ledge and separates into a three-pointed star and a triangle robot. The three-pointed star robot lands on top of the triangle robot. Next, the three-pointed star robot crawls off the triangle and reconnects to it, ultimately regaining the diamond-with-tail shape.

on the raised platform above the opening (see $t = 0$ min to 1:51 min). The raised platform mimics a washed-out canyon with overhanging walls that only leave a narrow gap at the top. A three-pointed star and a triangle then crawl underneath it. The three-pointed star connects to one of the triangle vertices by extending one of its links, as shown at $t = 0:30$ min. The ratchet tetrahedron can then reach down through the narrow gap, like a crane, connect to that same vertex

and lift it up (see $t = 4:58$ min to 8:28 min). Since the whole weight of another tetrahedron exceeds the holding power of the ratchet link's magnet connection, it has to support the ratchet Truss Link body on the edge of the gap (see $t = 7:01$ min). In this way, the links below can move around without risking the structural integrity of the ratchet above. The three-pointed star's two free links then shuffle their way toward the triangle's vertices, as shown from $t = 7:01$ min
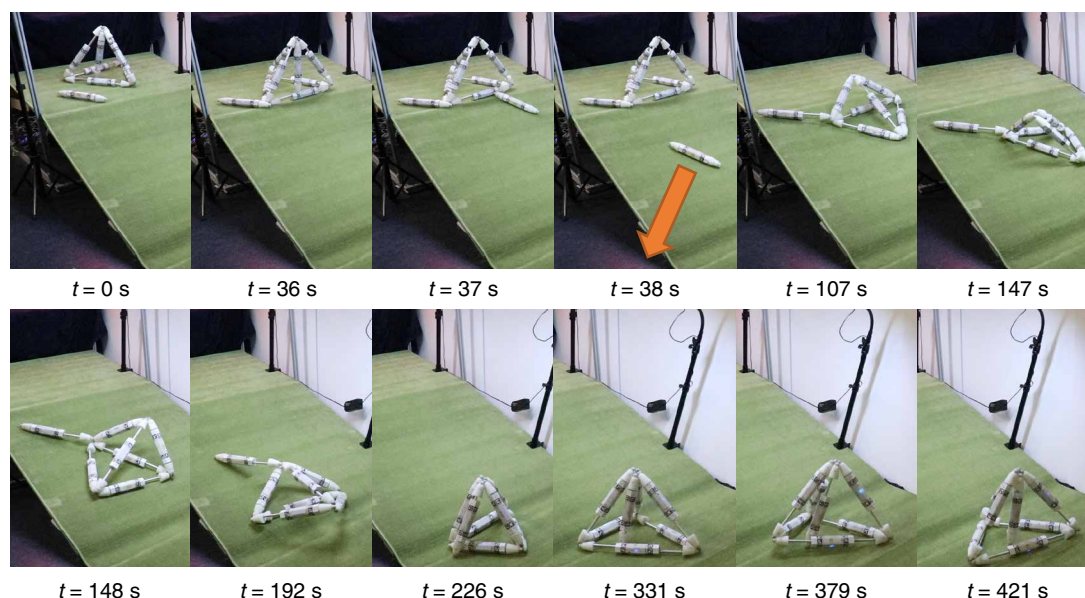
Fig. 8. A ratchet-tetrahedron sheds a "dead" ratchet Truss Link and picks up a replacement. The ratchet tetrahedron approaches the single Truss Link and latches onto it. Next, it sheds the dead link: The fully contracted and detached "dead" Truss Link falls off of the tetrahedron and rolls down the slope. The tetrahedron then topples itself twice to re-orient itself to pick up the newly found Truss Link. After the pickup at $t = 192$ s, the tetrahedron swings the Truss Link into its center and ratchets away.

to $t = 8:06$ min, until they connect; voila, a tetrahedron is formed. Next, the ratchet tetrahedron needs to disconnect from the tetrahedron. The ratchet tetrahedron drops the newly formed tetrahedron by fully contracting one side of its ratchet Truss Link and retracting the magnet inside the connector (see $t = 8:28$ min). Then, at $t = 9:57$ min, the newly formed tetrahedron crawls away. At this point, the next three-pointed star and triangle could come along and undergo the same assisted transformation.

We empirically explored various experiment setups and found over the course of 61 trials that a platform with a slot rather than a hole and a three-pointed star connected with one link to the triangle as the flat pattern facilitates the transformation. Using a diamond-with-tail topology as the starting point for the transformation was unsuccessful. After the last change to the experiment setup was made, three tetrahedrons were formed over 10 attempts. Common causes of failure were operator error leading to the ratchet tetrahedron collapsing and Truss Link malfunctions due to low battery or WiFi connectivity issues.

Through this experiment, we showed that the difficulty of forming a Truss Link tetrahedron can be reduced by robots assisting robots. This method of tetrahedron formation could be repeated without navigating the risks of folding an under-constrained three-pointed star by crawling it off a drop. Last, the transformation shown in this experiment aligns with the constraints of the robot metabolism and shows that robot development need not be a solitary endeavor.

DISCUSSION
We presented a robotic system that can produce structures that can develop physically, i.e., grow in size and capability, by absorbing and integrating found Truss Links or existing Truss Link structures. Many self-reconfiguring robotics systems have been demonstrated in the past, including our own systems capable of self-production (7). Unlike the Truss Link platform presented here, none of these

systems could develop from single 1D cells to a full 3D robot, while systematically improving its own capability in the process and without requiring external machinery.

Limitations and future work
The robot structures presented in this paper are very simple. This is a direct result of the still nascent stage of the field of self-reconfiguring modular robotics and the software infrastructure surrounding it. The Truss Link's design was deliberately kept to the bare minimum required to perform this demonstration. We believe that smaller and simpler building blocks will ultimately span a larger space of potential robot morphologies. However, practical considerations dictated by available linear actuators limit the expansion ratio, weight, and strength of each link in this study. In future work, we aim to develop microscale models that would allow the construction of single robots composed of millions of cells.

High cost and manufacturability, sensor integration, communication and control, and simulation are known challenges for modular robotic systems. Our high-fidelity simulation was sufficient to explore robot metabolism as a proof of concept, but it lacked the performance for machine learning–based control algorithms. A massively parallel, high-fidelity simulator would open the door to studying both design exploration and validation, sensor integration, and communication and control for the next generation of modular robots capable of robot metabolism. Thus, we plan to develop such a simulation environment for Truss Links in future work.

The Truss Link platform was designed around off-the-shelf components and built using commonly accessible tools to make it easy to replicate by anyone. With more than $200 in material cost per unit, Truss Links are neither cheap nor designed for mass production. A custom actuator design combined with a custom circuit board and a single battery power source could decrease the form factor significantly while still maintaining a high expansion ratio, enabling more impressive self-assembly results. In addition, a custom circuit could
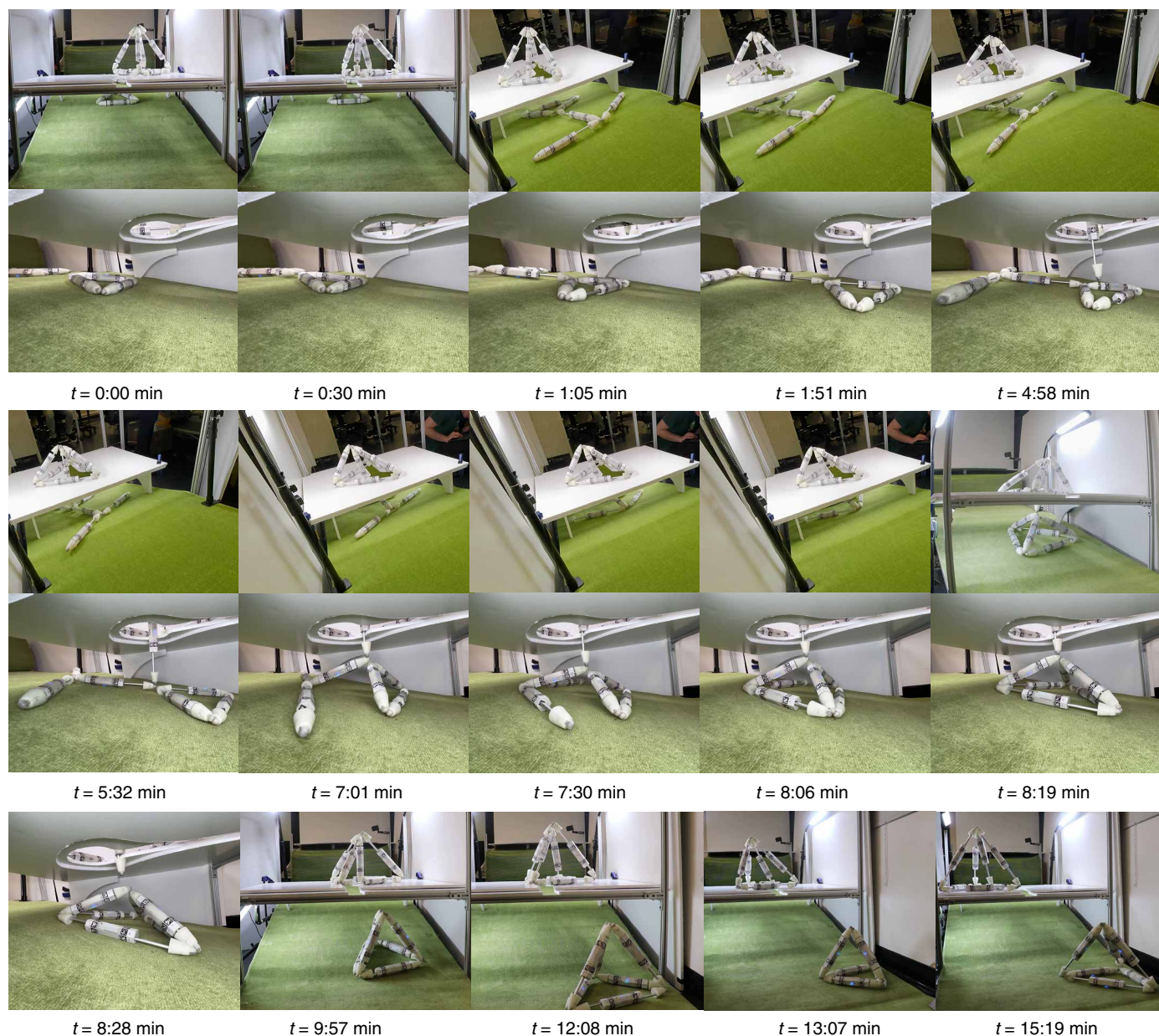
Fig. 9. A ratchet tetrahedron raises a 2D robot to become a tetrahedron robot. A ratchet tetrahedron uses its ratchet Truss Link to fish through a hole in the white platform for the vertex where the triangle and the three-pointed star are connected. After being lifted up, the three-pointed star connects to the two free vertices of the triangle, forming the tetrahedron. The different, time-synchronized camera angles in the frame sequence were picked based on which camera provided the most informative view of each stage.

integrate current sensing for the actuators, encoders, an inertial measurement unit (IMU), power management, and a charging circuit. This type of custom hardware, while promising more impressive results, would increase the cost of a single unit and make the research harder to replicate. We believe the promise of low-cost, mass-manufactured modular robots can only be achieved once academic research demonstrates a business case with a clear path to profitability for this technology and thereby sparks industry adoption.

Integrating sensors into the modules comes with the challenge of communicating and processing the sensor data. Sensors such as IMUs, magnetometers, current sensors, cameras, microphones, etc.,

can produce high-frequency, high-resolution time series data that would have to be processed onboard using an edge computing module instead of a simple microcontroller. Given this type of system, Truss Link modules could then be programmed to modulate their behavior in a decentralized fashion based on sensor readings while following a global objective.

Since Truss Link structures can change their topology, the controller must deal with changing kinematics and dynamics, as well as underactuated joints. First, we plan to explore a centralized control solution assuming perfect pose information of all Truss Links. Using search algorithms combined with reinforcement learning, we can

identify robot topologies and learn corresponding locomotion controllers. In addition, we can learn transition controllers that allow the robot to morph from one topology to another. Second, we plan to explore a decentralized control approach where each Truss Link's behavior is dependent on its sensor readings, local signaling between neighboring modules, and a globally shared objective. For example, by fusing force readings from actuators with IMU, and 360 camera readings, Truss Links could learn an end-to-end controller for self-assembly. In addition, Truss Links could be equipped with sonar, radio frequency, or light-based local signaling equipment as pathways for learned communication patterns. Ultimately, transferring these learned controllers from simulation to reality poses a significant challenge due to unaccounted-for differences between the simulated Truss Links and the physical system. For these reasons, we see the development of a high-fidelity, massively parallel simulation as the logical next step.

Applications for platforms capable of robot metabolism are distant but inevitable. As our economic welfare grows increasingly dependent on robots, it becomes necessary that these robots can take care of themselves physically. It is unlikely that human engineers will be able to maintain the growing numbers of robotic systems or manually adapt them to new needs, tasks, and environments, given their increasing complexity. We must understand how to build robot building blocks that enable robots that physically care for themselves, adapt, and grow. In essence, we need to create a self-sustaining robot ecology.

MATERIALS AND METHODS

Here, we share additional information on our experiment environment and how the walking speeds of different Truss Link topologies were measured. We provide in-depth information on the Truss Link's hardware and design. We explain how the Truss Links were coordinated using our Truss Link server and controller. Next, we address the key aspects of our customized PyBullet simulation environment used in our experiments. A rendered video of a diamond-with-tail forming in a randomized experiment can be seen in movie S4.

Experiment environment details

Our experiment setup (shown in fig. S1) was designed to allow the Truss Links to transition from single links to a ratchet tetrahedron. The experiment environment was designed with adjustable slopes for each stage. We initially set the slopes to the values that were used in the simulation and then adjusted the slopes as needed to achieve the transformations shown in the multistage robot self-assembly experiment. Stages one to four are 1.2, 0.6, 0.6, and 1.2 m long, respectively. The surface is built from 6-mm-thick plywood that is covered with a layer of 10-mm-thick foam board to smoothen the stage transitions from stages one to two and three to four. In addition, a cardboard cylinder containing weight was placed as an obstacle on stage three to allow the diamond-with-tail to fold itself into a tetrahedron in a controlled manner.

All our physical experiments were conducted on a 4-mm pile polypropylene carpet to ensure a consistent experiment surface. We used stationary cameras and light-emitting diode lighting to film each experiment.

Walking speed experiments

To assess the walking speed of different topologies during successful crawl cycles, we conducted a repeated locomotion experiment and plotted the results. Truss Links rely on differential friction for crawling and can get slowed down or stuck on uneven surfaces. Since this experiment aimed to assess speeds during successful walking or crawling maneuvers, we excluded video sequences where a topology got stuck on an uneven surface or crawled outside of the experiment setup from the measurement data. The speed measurements reported in this section were all collected on a 10° downward slope to mimic the conditions of stage four of the experiment setup. The gates used in the speed experiments were manually programmed, tuned on the basis of empirical observations, and then executed in an open-loop fashion. We marked the experiment surface with a line every 5 cm to track the robot speeds from the video footage.

The experiment results are shown in Fig. 4. The findings show that a crawling link, while only being able to move in a single dimension, is faster than a triangle. The triangle, which is superior to the individual Truss Link by being able to move in two dimensions, underperforms the single Truss Link's speed due to its increased weight and inopportune Truss Link angles. The crawling tetrahedron is slightly faster than a single Truss Link and demonstrates the most consistent performance. The ratchet tetrahedron is the fastest topology tested in this experiment but also the one with the most variance in speed. During its crawling motion, the ratchet tetrahedron tends to rotate and orient itself away from the slope direction, which causes it to slow down or move sideways rather than forward. This instability in the ratchet tetrahedron gate could be compensated for during closed-loop operation but was included intentionally to reflect the raw system's dynamics.

Truss Link design

The Truss Link is the homogenous building block of our truss-type modular robot system. Truss Links allow for the construction of chain and lattice structures. The main hardware innovation is the Truss Link's compliant magnetic connector that passively orients the polarity of a 1.27-cm-diameter neodymium magnet sphere inside the connector to generate an equilibrium of attraction among all modules trying to connect at a single point. According to our in-line dynamometer pull-away tests, two connectors require a pull-away force of approximately 13.7 N to be separated. Modular robot designs commonly incorporate communication channels into their connectors (27). We opted not to use the connector for power sharing or communication to reduce the design complexity and increase the connector's versatility. Our design can form connections without needing passive connector blocks, such as the ones used in Morpho or the Odin robot, since that would have complicated self-assembly (17, 18).

We designed the Truss Link platform to form a tetrahedron structure capable of picking up a Truss Link attached to a base vertex by toppling itself over. To achieve this motion, the tetrahedron must be able to sufficiently shift its center of mass without collapsing. A geometric analysis revealed that the Truss Link's minimum expansion ratio, the maximum length of expansion a Truss Link can achieve as a percentage of the minimum length of a link, must be more than 41.5% to allow for the tetrahedron toppling behavior. Our current Truss Link design with a contracted length of 28 cm and an expanded length of 43 cm achieves an expansion ratio of 53%.

Each Truss Link body comprises two prismatic actuators, one particle photon microcontroller, a WiFi antenna, a voltage regulator, a voltage divider, and batteries. As our actuator, we chose the 100 mm stroke length Actuonix L-12I linear servo with a gearing ratio of 210:1. Its small form factor and simple control interface facilitated

integration. Since the motor housing of the linear servo is the same size for each stroke length, we maximized the Truss Link's expansion ratio by picking the Actuonix L-12 servo model with the maximum stroke length.

The two linear actuators can be both independently and jointly actuated. The Truss Links were designed with a passive attachment/detachment mechanism in each connector. Considering the detachment mechanism as a separate degree of freedom (DoF), each Truss Link is a 4-DoF system. Aside from the replacing a "dead" Truss Link, and robot-to-robot assisted reconfiguration experiments, we treat each Truss Link as a 2-DoF system since the attachment/detachment mechanism is not used.

Truss Links are powered by two removable single-cell 380-mAh lithium polymer batteries that are connected in series. We step down the voltage to 5 V for the particle photon via a voltage regulator and use a voltage divider to monitor the battery voltage via the onboard 12-bit analog-to-digital converter. Please refer to the Supplementary Materials for further technical details regarding the Truss Link system.

Truss Link connector

The Truss Link uses a free-form magnetic connector with a detachment mechanism. The connector comprises Fused Deposition Modeling (FDM)–printed body shells and a magnet holder, as well as a 12.7-mm-diameter N52 neodymium magnet sphere, a conical compression spring, two screws, and two heat-set inserts (see Fig. 2C). The entire connector is held in place via the magnet holder, which is screwed and hot-glued directly into the servo shaft. The magnet holder constrains the magnet's position while allowing it to rotate freely, so it can align its polarity when connecting with other connectors. To reduce friction during magnet alignment, we apply a dry graphite lubricant on the inside of the magnet holder.

The connector detaches by retracting the magnet inside the connector shell, thereby reducing the magnetic field outside the connector. The magnet gets retracted by 1.2 times its diameter to ensure secure detachment. The connectors' shape results from a tradeoff between optimizing magnet connection strength while maintaining the ability to disconnect via the magnet pullback mechanism. The small rim below the connector tip prevents other connectors from simply sliding along the outside of the connector during detachment, while the conical shape helps create distance between the magnet sphere and any outside connectors.

The connector detaches by fully retracting the Truss Link's servo, thereby pushing the connector shell against the center body, and retracting the magnet holder. When expanding after a full retraction, the magnet holder resets itself back into an attachment-ready position using the conical spring inside the connector (see Fig. 2, C and D). To ensure a smooth resetting behavior, we smoothen and grease the friction points between 3D printed parts.

To minimize the center body size, the servos were aligned in parallel but in opposite directions. As a result, the servo shafts are not centered on the Truss Link body's central axis but placed next to each other. To compensate for this offset, we added an offset to the magnet holder, such that the connector tips are centered on the body's central axis. Centering the connector on the link's central axis allows us to balance the geometry reduce rotations along the link axis during maneuvers, and thereby improving the stability Truss Link structures.

We noticed during experiments that connectors and servo shafts sometimes come loose after repeated use and then rotate out of alignment. One possible solution could be to retrofit the servo motors with square shafts and square shaft guides. This would prevent both the connector shell and shafts from rotating and unscrewing themselves.

Supplementary Materials

The PDF file includes:
Supplementary Text
Figs. S1 to S4
Table S1
Legends for movies S1 to S5
Legend for data S1

Other Supplementary Material for this manuscript includes the following:
Movies S1 to S5
Data S1

REFERENCES AND NOTES

1. L. Von Bertalanffy, The theory of open systems in physics and biology. *Science* **111**, 23–29 (1950).
2. E. Crespi, R. Burnap, J. Chen, M. Das, N. Gassman, E. Rosa, R. Simmons, H. Wada, Z. Q. Wang, J. Xiao, B. Yang, J. Yin, J. V. Goldstone, Resolving the rules of robustness and resilience in biology across scales. *Integr. Comp. Biol.* **61**, 2163–2179 (2022).
3. J. M. Reed, B. E. Wolfe, L. M. Romero, Is resilience a unifying concept for the biological sciences? *iScience* **27**, 109478 (2024).
4. I. A. Kanaev, Evolutionary origin and the development of consciousness. *Neurosci. Biobehav. Rev.* **133**, 104511 (2022).
5. C. A. Lage, D. W. Wolmarans, D. C. Mograbi, An evolutionary view of self-awareness. *Behav. Processes* **194**, 104543 (2022).
6. S. Kriegman, D. Blackiston, M. Levin, J. Bongard, Kinematic self-replication in reconfigurable organisms. *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2112672118 (2021).
7. M. Akram, H. M. S. Asif, M. Uzair, N. Akhtar, A. Madni, S. M. A. Shah, Z. U. Hasan, A. U. N. Available, Amino acids: A review article. *J. Med. Plant Res.* **5**, 3997–4000 (2011).
8. P. M. Wyder, R. Bakhda, M. Zhao, Q. A. Booth, S. Kang, M. E. Modi, A. Song, J. Wu, P. Patel, R. T. Kasumi, D. Yi, N. N. Garg, S. Bhutoria, E. H. Tong, Y. Hu, O. Mustel, D. Kim, H. Lipson, "Robot Links: Towards Self-Assembling Truss Robots," *2024 6th International Conference on Reconfigurable Mechanisms and Robots (ReMAR)*, 525–531 (2024).
9. T. Fukuda, S. Nakagawa, "Dynamically reconfigurable robotic system," in *Proceedings. 1988 IEEE International Conference on Robotics and Automation* (IEEE Comput. Soc. Press, 1988), pp. 1581–1586.
10. M. Yim, P. White, M. Park, J. Sastra, "Modular self-reconfigurable robots" in *Encyclopedia of Complexity and Systems Science* (Springer New York, New York, NY, 2009), pp. 5618–5631.
11. K. Gilpin, D. Rus, Modular robot systems. *IEEE Robot. Autom. Mag.* **17**, 38–55 (2010).
12. V. Zykov, E. Mytilinaios, B. Adams, H. Lipson, Self-reproducing machines. *Nature* **435**, 163–164 (2005).
13. K. Stoy, D. Brandt, D. J. Christensen, *Self-Reconfigurable Robots: An Introduction* (MIT Press, 2010).
14. A. Spinos, D. Carroll, T. Kientz, M. Yim, "Variable topology truss: Design and analysis," *IEEE International Conference on Intelligent Robots and Systems* 2017-September, 2717–2722 (2017).
15. E. Park, J. Bae, S. Park, J. Kim, M. Yim, T. W. Seo, Reconfiguration solution of a variable topology truss: Design and experiment. *IEEE Robot. Autom. Lett.* **5**, 1939–1945 (2020).
16. H. Yoon, J. Bae, H. Li, T. Seo, M. Yim, "Compliant Spherical Joint Design for Reconfiguration of Variable Topology Truss," *2024 6th International Conference on Reconfigurable Mechanisms and Robots (ReMAR)*, 500–505 (2024).
17. C. H. Yu, K. Haller, D. Ingber, R. Nagpal, "Morpho: A self-deformable modular robot inspired by cellular structure," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IROS (IEEE, 2008; http://ieeexplore.ieee.org/document/4651130/), pp. 3571–3578.
18. A. Lyder, R. F. M. Garcia, K. Stoy, "Mechanical design of Odin, an extendable heterogeneous deformable modular robot," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IROS (IEEE, 2008; http://ieeexplore.ieee.org/document/4650888/), pp. 883–888.
19. R. F. M. Garcia, A. Lyder, D. J. Christensen, K. Støy, "Reusable electronics and adaptable communication as implemented in the Odin modular robot," *2009 IEEE International Conference on Robotics and Automation*, 1152–1158 (2009).
20. P. Grouchy, H. Lipson, "Evolution of self-replicating cube conglomerations in a simulated 3D environment," *Artificial Life 13: Proceedings of the 13th International Conference on the Simulation and Synthesis of Living Systems*, ALIFE 2012, 59–66 (2012).

21. C. Liu, Q. Lin, H. Kim, M. Yim, SMORES-EP, a modular robot with parallel self-assembly. *Auton Robots* **47**, 211–228 (2023).

22. V. Zykov, P. Williams, N. Lassabe, H. Lipson, "Molecubes extended: Diversifying Capabilities of Open-Source Modular Robotics," in *IROS-2008 Self-Reconfigurable Robotics Workshop* (2008).

23. J. W. Romanishin, K. Gilpin, S. Claici, D. Rus, 3D M-Blocks: Self-reconfiguring robots capable of locomotion via pivoting in three dimensions. *2015 IEEE International Conference on Robotics and Automation*, Seattle, WA, 26 to 30 May 2015 (IEEE; 2015), pp. 1925–1932.

24. 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)*, 4239–4245 (2022).

25. 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)*, 6506–6513 (2020).

26. H. Lipson, J. B. Pollack, Automatic design and manufacture of robotic lifeforms. *Nature* **406**, 974–978 (2000).

27. R. Fitch, K. Stoy, S. Kernbach, R. Nagpal, W.-M. Shen, Reconfigurable modular robotics. *Rob. Auton. Syst.* **62**, 943–944 (2014).