This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LRA.2017.2716418, IEEE Robotics and Automation Letters

1D Printing of Recyclable Robots

Daniel Cellucci[1], Robert MacCurdy[2], Hod Lipson[3], and Sebastian Risi[4,*]

***Abstract*—Recent advances in 3D printing are revolutionizing manufacturing, enabling the fabrication of structures with unprecedented complexity and functionality. Yet biological systems are able to fabricate systems with far greater complexity using a process that involves assembling and folding a linear string. Here, we demonstrate a 1D printing system that uses an approach inspired by the ribosome to fabricate a variety of specialized robotic automata from a single string of source material. This proof-of-concept system involves both a novel manufacturing platform that configures the source material using folding and a computational optimization tool that allows designs to be produced from the specification of high-level goals. We show that our 1D printing system is able to produce three distinct robots from the same source material, each of which is capable of accomplishing a specialized locomotion task. Moreover, we demonstrate the ability of the printer to use recycled material to produce new designs, enabling an autonomous manufacturing ecosystem capable of repurposing previous iterations to accomplish new tasks.**

***Index Terms*—Assembly, Mechanism Design, AI-Based Methods, Neural and Fuzzy Control**

I. Introduction

WHILE advances in 3D printing have allowed robot mechanisms to be produced with greater ease and speed [1], and new additive manufacturing materials and processes are beginning to enable on-demand printed circuits [2]–[4], the 3D printing of complete systems that include actuation and energy storage is still in its infancy. The potential of this design and manufacturing scheme has not yet been leveraged to fabricate complete robots; they are still manually designed and constructed, a complex, time-consuming process that requires experts at all stages. The major goal of a robot "walking out of the printer" is not realizable with current additive manufacturing technology. Additionally, utilizing *recycled material* is infeasible with virtually all 3D printing methods. Though in principle a 3D-printed object made from a single material could be ground down into its base material and reused, the facilities required to perform this operation make it impractical for robot applications in remote and inaccessible locations (where an on-demand and reusable robot fabrication system would be particularly useful). Also, this recycling approach is inapplicable to robots fabricated from multiple materials.

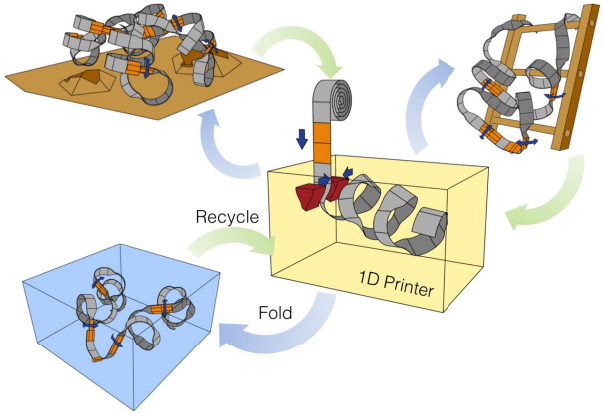

Fig. 1: **1D Robot Manufacturing System.** Our 1D printer is able to fold specialized robots for different tasks on-demand, all derived from a single generic strip of deformable source material (shown in gray) with *pre-embedded* actuators and control elements (shown in orange). The system is able to reuse the *same* material to fold a different three-dimensional robot for a different task by unfolding the deformable parts of the material and feeding the strip back into the printer.

To address these challenges, we present a proof-of-concept fully-automated design and assembly process, inspired by the ribosome, that can automatically discover solutions to high-level design challenges and instantiate the designs as physical artifacts. Our contributions in this work outline our 1D robot fabrication concept, demonstrate the technical aspects required to implement the printer and its source material, and describe the theoretical details of the optimization process used to design the robots. In previous work (GOLEM) we showed how evolutionary methods could be used to design simple moving robots that were partially fabricated automatically [5]. Others have aided the robot design process by allowing a human designer to compose modular subsystems [6], [7], or shown how specific pre-designed robots can self-fold [8]. Here, we demonstrate a complete, autonomous system that synthesizes designs from high-level behavior specifications and then automatically fabricates ready-to-use robots.

Our system employs an evolutionary-based approach to discover the sequence of folds required to create a specialized robot out of a 1-dimensional strip of material. The *pre-manufactured* source material has been augmented with

Manuscript received: Feb., 16th, 2017; Revised May, 18th, 2017; Accepted May, 24th, 2017.

This paper was recommended for publication by Editor John Wen upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the NASA Space Technology Research Fellowship (NNX13AL38H) awarded to D. Cellucci, and by a National Science Foundation Graduate Research Fellowship (DGE-0707428) awarded to R. MacCurdy. Fig. 5 was generated with support from the NASA STRG Game Changing Development Program.

[1]D. Cellucci is with the Department of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY, 14850 USA.

[2]R. MacCurdy is with the Computer Science and Artificial Intelligence Lab, MIT, Cambridge, MA 02139, USA

[3]H. Lipson is with the Department of Mechanical Engineering, Columbia University, New York, NY 10027, USA

[4]S. Risi is with the IT University of Copenhagen, Copenhagen, Denmark. e-mail: sebr@itu.dk (*Contact Author)

Digital Object Identifier (DOI): see top of this page

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LRA.2017.2716418, IEEE Robotics and Automation Letters

actuators and other control elements and is folded into the prescribed configuration by a custom "printer" (Fig. 1 and 3). This process forms the basis of a generalized method for automatically creating robots tailored to a particular task. Because these robots use the same starting source material, they can be easily recycled into a different design when no longer needed. In this paper we demonstrate how our method can automatically design and fabricate robots for three different locomotion tasks, and how a robot designed for one task can be recycled into another.

Previous examples of folding applied to robotics include strings that self-configure into complex structures passively through magnets [9] or via electrical motors [10], [11], origami-inspired systems that generate 3-dimensional robots from two-dimensional planes with actuated hinges [8], [12], [13], 3D-printed objects that fold in response to temperature [14] or humidity changes [15], and machines that can manufacture non-actuated mechanisms from a flat ribbon [16] or filament [17]. Printing integrated electromechanical systems that include sensing, computation, actuation and energy storage is a persistent challenge for approaches that build with "raw materials" like plastic filament or conductive paste. Robot systems based on prefabricated modular designs sidestep these challenges [18]–[20], though they share one or more common drawbacks, including relatively large module size, high complexity and cost, as well as module-interconnect challenges. The complexity, strength, and cost of electromechanical connections between modules has been specifically identified as an ongoing issue [21], [22], and we point out that our approach was chosen to avoid these problems. Additionally, in our approach the overall size and power consumption of the resulting robots is not determined by the angle-holding torque of an individual module (a challenge noted in [11]).

Our work extends previous efforts to automatically fabricate robots by relaxing the requirement that the machine *self-reconfigure*, placing that capability in a dedicated assembler instead. Doing so removes complexity (and associated energy, cost and size implications) from the fabricated robots. In contrast to self-folding approaches that either require dedicated hardware at every possible fold-site [10]–[12] (regardless of whether any particular design uses the fold or not), or are designed for one particular robot [8], [13], our approach relies on the reversible deformation of a material (metal wire) that is low-cost, readily mass-produced, and can create a multitude of designs with no human effort. A related system was developed by Brodbeck et al. [23], where a robot arm assembles modular robots. However, our 1D printing approach allows customized, application-specific geometries to be obtained with fewer discrete modules and, consequently, with less structural complexity and cost.

The use of an external fabrication apparatus to impart a particular desired structure onto a generic input material is inspired by the approach taken by biological systems during protein synthesis. The ribosome enables the construction of myriad chemicals that form the basis of all cells through the ordered sequential assembly of amino acids [24]. In particular, the ribosome also plays a role in determining the ultimate morphology of a protein as it is assembled by modulating the synthesis rate, which impacts the folding pattern [25]. The ribosome is clearly distinct from the approach explored here. However, adopting the use of an external folding mechanism allows a simple, generic, linear input material to be converted into a variety of special-purpose robots.

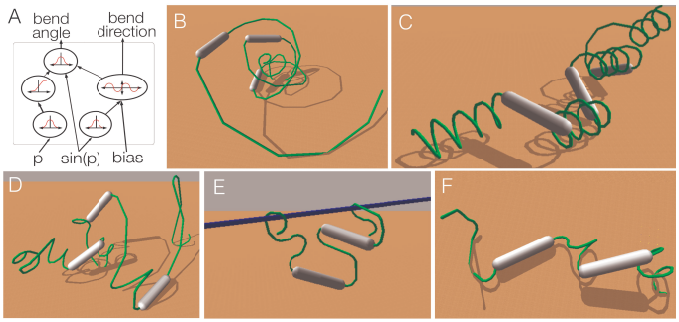

Fig. 2: **Robotic Representation and Example Designs.** The folding pattern of a particular robot is determined by a compositional pattern producing network (CPPN) (**A**). An optimization algorithm searches through increasingly complex CPPNs to find robot designs that satisfy specific high-level goals; this approach allows a variety of three-dimensional robots to be produced from a one-dimensional strand of material (**B**–**F**). To instantiate a robot design, the CPPN is queried sequentially at fixed intervals with the current segment number p as input, and the desired bend direction and angle as output. In addition to the fold directions, the CPPNs also determine control parameters for each motor module.

II. Robot Representation

Our system creates robots by sequentially folding a 1-dimensional wire until the desired 3-dimensional robot is formed. The wire contains pre-embedded actuators at fixed intervals, allowing different segments of the wire to rotate relative to each other. These actuators communicate wirelessly, allowing coordinated motion control. Each motor module can apply a rotation in the interval [-90, 90] degrees between the two wires that are connected to either end of it.

The pattern of folds in the wire is encoded by a modified version of a *compositional pattern producing network* (CPPN; [26]), a special type of artificial neural network. CPPNs are inspired by evolutionary development, and are able to create complex artifacts such as two-dimensional images [27], three-dimensional forms [28], and connectivity patterns of high-dimensional neural networks [29]. While they have been used previously to encode morphologies of simulated robots [30], [31], here we demonstrate for the first time the transfer of CPPN-encoded robots to the real world. The key concept behind CPPNs is that they generate a solution to a problem by iteratively composing more primitive functions in a directed graph, adding functions and weighted connections to the graph until a satisfying solution is achieved. The resulting functional representation is *generative*: it does not require as many internal parameters as the morphology of the object that it defines would dictate. Imposing this structure on the representation of the object dramatically reduces the dimensionality of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LRA.2017.2716418, IEEE Robotics and Automation Letters

the search, making large, complex problems tractable. In the work here, the CPPN for each robot was applied along the length of the robot's body (Fig. 2A), generating the sequence of folds required to describe its morphology. Our CPPN-encoding utilized activation functions with regularities such as as symmetry (e.g. Gaussian) and repetition (e.g. sine) to facilitate the discovery of robot designs that satisfy the desired behavior specifications (Fig. 2B-F). Importantly, CPPNs also allow regularities with variation [26], which is challenging for more regular indirect encodings such as L-systems [32].

The inputs to the CPPNs (Fig. 2A) are the current segment number p scaled to [-1,1], and $(sin(p)+1)/2$, which facilitates the evolution of structures with repeating patterns. The z-rotation output (*bend angle*) determines the rotation of the bend head, while the second output b (*bend direction*) determines the direction of the fold (+30 degrees if $b < 0.0$, -30 degrees otherwise). If a fold would result in a collision of the wire with the printer, the fold is skipped and the wire is simply advanced forward (fed out of the printhead) one segment length. In addition to the morphological description, two extra CPPN outputs (not shown in Fig. 2A) encode the motor control signals, automating the motor controller-design task. These two outputs determine the amplitude A and phase φ for a modified sine wave motor-activation function: $A\ sin(t + \varphi)$. For each motor with position p on the string, the CPPN is queried at that location to determine the specific amplitude and phase values that control the angle of each motor module. φ is scaled to $[-\pi/2, +\pi/2]$.

III. Optimization

We employ a multi-objective evolutionary computation approach [33] to optimize a set of CPPN-encoded robots in a 3D rigid-body physics simulation using the freely available Open Dynamics Engine. Controlled tests allowed the simulation parameters (e.g. friction, maximum motor torques and speeds, material density) to be calibrated in order to minimize the difference between the behaviors of the robots in simulation and in reality. Evolutionary algorithms have shown promise in solving complex engineering tasks with multiple competing objectives and large numbers of decision variables [34]. They have also been used successfully for different robot design tasks [35], motivating their application here. The potential design space of our robots, composed of N individual segments, each of which can be bent either up or down by 30° in 180 different orientations, is $2^{N-1}180^{N-1}$. Typical robot designs assign $N \in [50...150]$, rendering an exhaustive brute-force search infeasible (the number of possible configurations exceeds the estimated computational capacity of the universe [36]). In contrast, the search space explored by the CPPN-based approach increases gradually during evolution, and the complexity of the final representation does not typically exceed 50 connections. For example, the CPPN encoding the automaton in Fig. 2E has 30 connections and the automaton in Fig. 2F has 50 connections.

The CPPNs are optimized for specific tasks by the *Neuroevolution of Augmenting Topologies* (NEAT) algorithm [37], [38], which can evolve neural networks and therefore also CPPNs. The initial population in NEAT is composed of random CPPNs in which the network inputs are directly connected to its outputs. NEAT then adds connections and nodes over the course of evolution, making them more complex. The size of the network does not need to be set a priori; because NEAT "grows" candidate solution networks, it avoids unnecessarily searching through high-dimensional solution spaces when a simpler solution is adequate.

The fitness of each robot is determined by (1) maximizing its speed in the specified domain, (2) maximizing the "compactness" of the produced design, and (3) minimizing the bending-torque required during folding. The process iteratively selects fitter machines, creating offspring by modifying the underlying CPPN description via mutation and crossover between fitter individuals in the population. To encourage the evolution of a diverse population of designs, we employ the popular multi-objective optimization approach NSGA-II [33] together with *novelty search* [39], [40]. Novelty search offers a more exploratory and divergent evolutionary search than traditional objective-based methods and augments the fitness function with a novelty metric that rewards diverse phenotypes. In this paper, the novelty p of an individual x is measured in morphological space as given by: $\rho(x) = \frac{1}{k}\sum_{i=1}^{k} \text{dist}(x, \mu_i)$, where μ_i is the ith-nearest neighbor of x with respect to a distance metric $dist$. The metric $dist$ is the average euclidean distance between the vectors of folds that describe different robots. If the novelty is above a threshold ρ_{min}, then the individual is entered into a permanent archive. When the novelty of an individual is computed, it is calculated by finding the k-nearest neighbors in the joint set of individuals in the archive and the current population. In this work $k = 15$.

In addition to novelty, the second objective in our NSGA-II approach is a traditional fitness function that rewards individuals for traveling as far as possible in the allotted amount of time: $T = |p_s - p_e|\ (1.0 - tq)\ c$, where p_s is the starting position of the robot and p_e is the ending position after the evaluation period. To facilitate the evolution of designs that are within the design space of our printer, the fitness function tries to minimize tq, which is the maximum torque on the design during the folding process, and maximize c, the compactness of the design (measured as: 1.0 - average distance over all segments to the center of mass of the robot). While some collisions during the folding process can be avoided by skipping a particular fold, collisions with the printer can also occur while the material is simply being fed through the machine (e.g. if a part of the robot that has already been bent is caught behind the printhead when the wire is advanced). Therefore designs are also rewarded for not colliding with the printer by adjusting the multi-objective fitness:

$$F_1 = \frac{T}{1 + cl}, F_2 = p, \tag{1}$$

where cl is the number of collisions during the simulated folding process. While the fitness function could be modified for each of the three navigation tasks (crawling, pipe traversal, and rolling), as is typical in evolutionary computation [34], we found that only scaling the CPPN outputs (Section II) to slightly different ranges facilitates the evolution of high-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LRA.2017.2716418, IEEE Robotics and Automation Letters

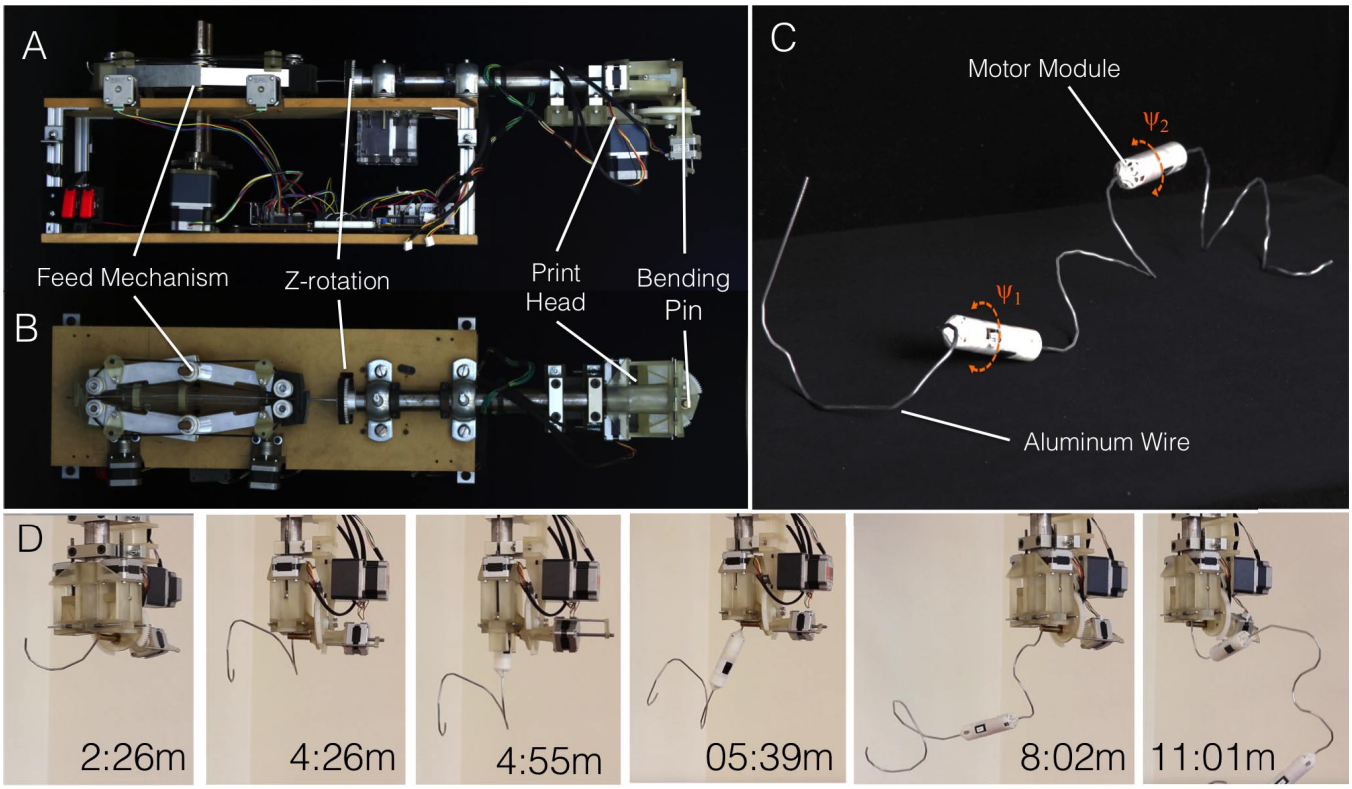

Fig. 3: **Hardware Demonstration**. Shown is a side (**A**) and top view (**B**) of the 1D robot printer. The feed mechanism and printhead allow variable-radius source material (i.e. motor modules connected to wires) to be transported through the machine (**C**) that can be transformed into different 3D robot morphologies. Once the robot is printed, it is detached from the machine and can actuate its motors, allowing it to navigate. The pre-embedded motor modules can rotate in the interval $[-90, 90]$ degrees (orange arrows). An example of the folding of a three-dimensional robot is shown in (**D**).

performing solutions. For the walker and roller A is scaled to the range $[0, \pi]$, while it is scaled to $[0, 1]$ for the pipe-traverser to prevent the robot from swinging violently and potentially falling off the pipe (since this possibility was not modeled during simulation). Motor angles for the crawler and pipe-traverser are limited to ± 45 degrees. For the roller, both motors follow an identical sine function (A and φ are determined by just querying the CPPN for the first motor position) and the motor limits are set to ± 90 degrees, which encourages the evolution of designs that locomote by rolling.

Optimization Parameters. The size of each population is 100 with 10% elitism. The number of generations is set to 300. Sexual offspring (75%) do not undergo mutation. For asexual offspring (25%), the probabilities of link weight mutation, link addition, and node addition are 0.75, 0.1 and 0.05, respectively. The available CPPN activation functions are sigmoid, Gaussian, absolute value, cosine, and sine, all with equal probability of being added. Parameter settings are based on prior reported settings for NEAT [37], [38].

IV. 1D Robot Printing System

The input material for our printer is 3 mm diameter 1100-alloy aluminum wire with motor modules embedded at regular intervals along its length. The wire has been treated with a soft (bend-and-stay) temper which minimizes bending error. Each motor module contains a Hitec HS-5065MG+ servo, a 100 mAh battery, and an MSP430-RF2500T wireless microcontroller. An acetal Delrin homopolymer housing encloses these components and connects to the two adjacent aluminum wires via shaft collars at each end. This housing is a cylinder, 26 mm in diameter, with conical endcaps and an overall length of 110 mm. The total mass of each motor module, including batteries and components, is 62 g.

The aluminum wires that mechanically connect adjacent motor modules do not provide any electrical connectivity. Since the clocks in each module are independent, a synchronization mechanism is required in order to ensure that the movements of every motor are coordinated. The wireless microcontrollers in each motor module provide this synchronization via a master-slave scheme: one motor module is used to provide the clock for all the other modules. The master periodically broadcasts a message that includes the absolute time. When a slave module hears this message, it sets its local clock and continues the motor command playback sequence. Each motor module has its own motor sequence, a function that maps time into motor position, which is determined by an evolved CPPN. These sequences are transferred wirelessly to each motor module.

The design of the printer that folds the material into the target morphology (Fig. 3A,B) was informed by the D.I.Wire Bender (www.instructables.com/id/DIWire-Bender/), which we customized and augmented with three mechanisms designed to accommodate variable-radius material. The first

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LRA.2017.2716418, IEEE Robotics and Automation Letters

is a wire-feed mechanism with two sets of arms that close around the wire to grasp it, or open to let a motor module pass through. The second is a print-head with a sliding door through which the material passes. The door can open to allow a motor module to pass, or close to grip the wire. Finally, the printhead also has a rotational degree of freedom, allowing it to rotate around the material's feed axis. These components are explained in more detail next.

The **feed mechanism** consists of two sets of identical arms with knurled steel drive cylinders mounted on the ends. The drive cylinders grip the wire tightly when the arms are closed, and are rotated by a single stepper motor using a set of gears and timing belts. The arms are spaced far enough apart that a motor mount can fit between the drive cylinders, allowing a motor to pass through the mechanism by opening and closing the arms in a specific sequence, which is illustrated in the attached video. In addition, the ends of the arms are fitted with alignment flanges, which recenter the wire onto the drive cylinders in the event of a misalignment.

The **printhead** consists of a spring-loaded door mechanism driven by two threaded stepper motors. This door is designed to provide clamping force during the bending step, while still opening wide enough to admit a motor module. Attached to the underside of this printhead is another stepper motor that drives the bending pin (a brass cylinder) into contact with the deformable wire. The bending pin is mounted on a lead-screw driven by a stepper motor, allowing it to retract far enough to allow a motor module to pass or to move under the wire to perform bends in the opposite direction.

In order to ensure that the printed robots accurately represented the simulated designs, we calibrated the printer to perform $\pm 30^\circ$ bends; the error at 30° was 2° with a standard deviation of 0.84°. The largest source of error was bend-back in the wire, in which the elastic deformation of the material produced a bend a few degrees smaller than desired, and the largest source of variation was due to irregularities in the linearity of the unbent wire. We found bending 2° past the target angle reduced the error due to bendback.

The **Z-rotation** degree of freedom allows the printhead to rotate around the feed axis, enabling bends to occur in any plane parallel to and intersecting this axis, which permits the formation of complex three dimensional structures from a succession of 2D bends. The circular cross-section of the wire simplifies the bending process, since the same contact surface is encountered regardless of z-rotation. However, this cross-section allows the wire to rotate slowly within the machine while it is being fed. As a consequence, if there is sufficient moment applied from the bent part of the wire, the printer can lose track of the θ-position (the wire slowly rotates as it is fed). In order to mitigate this moment we mounted the printer vertically, aligning the feed direction with the gravity vector. This reduced the impact of gravity on drift in the θ-position of the part; however, gravity does limit the maximum length of a cantilevered segment, as discussed in Section V-A.

V. Printing Examples

The printer (Fig. 3) produced ten complete robots in the course of development and testing; four worked as designed. Initial failure cases were due to mismatches between simulation and reality, as well as miscalibrations of the printer's bend angles. The input material for the examples shown in this paper all had an overall length of 2 meters (87 segments) with two motor modules embedded at 0.76 m intervals (Fig. 3C), subdividing the wire into three equal-length sections. Folding an 87-segment design takes approximately 13 minutes (Fig. 3D). When a robot is complete it is removed from the printer (the only manual step) and the evolved controller is executed by the motor modules to actuate the robot.

TABLE I: **Performance Results.** Shown are the number of motor cycles each robot required to travel its full length. The virtual models require approximately the same number of motor cycles as the physical robots to travel one body length in the same direction. The larger discrepancy in the performance of the pipe-traversal robot is mainly due to the difficulty in modeling friction between the robot and the pipe.

	Cycles/Body Length Traveled	
Morphology	**Virtual**	**Physical**
Crawler (Fig. 4B, 4D)	29	32
Pipe-traversal (Fig. 2E, 4E)	17	42
Roller (Fig. 2F, 4F)	0.75	0.37

Figure 4 shows three printed physical robots in action. The crawler robot (Fig. 4D) moves by using its two motors to rock its center of gravity forward and backward. On the forward cycle the rear portion of the robot loses contact with the ground, allowing its motor to move it forward. On the rearward cycle the rear portion is used to push-off and the cycle repeats. The pipe traverser (Fig. 4F) employs a related strategy; by fully encircling the pipe, the rear section of the design allows the robot to grip the pipe, which allows the front portion to lift off of the pipe and slide forward. This forward momentum moves the rear portion of the robot forward in a dynamic motion (this dynamic motion is more difficult to accurately model, possibly explaining the larger discrepancy noted in Table I). The roller robot (Fig. 4E) travels by alternately rotating its outermost segments while bracing against the floor with the opposing segment (note that this is *not* a wheel; the motors do not rotate through 360°). While the roller terrain in Fig. 4E appears uneven, these are artifacts of the backdrop used in the images and do not appreciably affect the robot's movement. A video showing the printer and the robots it produced is available here: https://youtu.be/ElW0O2IiuXA.

We tested the recyclability of the manufacturing platform by manually unfolding the crawler robot (i.e. straightening the aluminum wire while leaving the motors attached; automating this step is straightforward) and feeding it back into the printer to produce a robot that locomotes in a different manner (Fig. 4E). Both the recycled and non-recycled designs closely resembled their virtual counterparts in terms of morphology and locomotion behavior.

Because our approach allows robots to be fully recyclable, it could complement methods that employ evolution directly in the physical world. A purely real-world [23] or hybrid approach [41] could be useful for more complex robots, in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LRA.2017.2716418, IEEE Robotics and Automation Letters

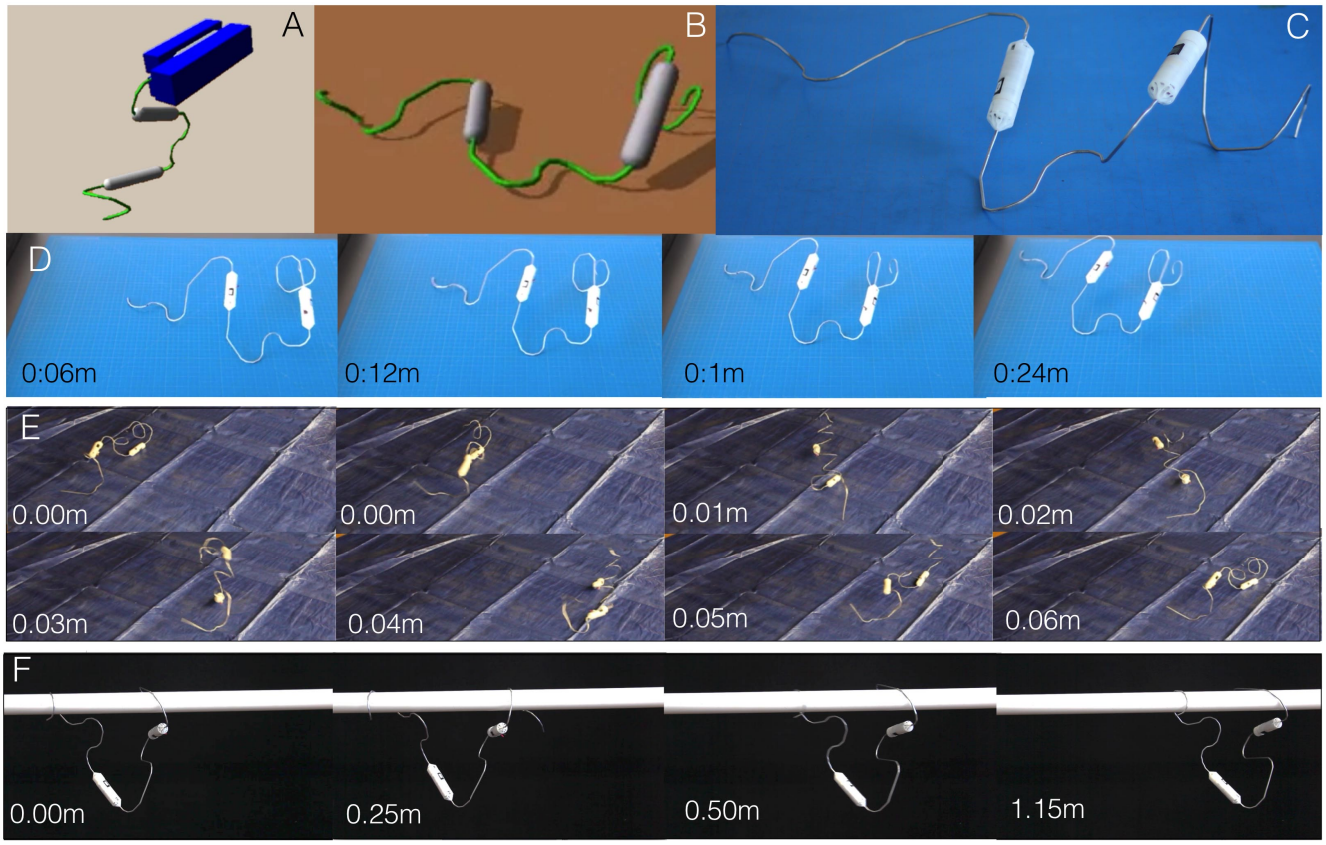

Fig. 4: **Robot Crawler, Roller, and Pipe-Traverser**. Designs folded by a simulated printer (**A**), are optimized to perform a certain task (**B**), and then transferred to reality (**C**). The printer is able to produce robots tailored to different locomotion strategies, such as a crawler (**D**), a roller (**E**) and a pipe-traverser (**F**). Video: https://youtu.be/ElW0O2IiuXA. Because the pre-manufactured source material is the same for these robots (i.e. motor modules are embedded at the same intervals), it can be reused to create new designs. To produce the roller (**E**), the crawler (**D**) was flattened out (i.e straightening the wire but leaving the motors attached), fed back into the printer and refolded. The virtual pipe-traverser and roller are shown in Fig. 2E,F.

which there might be a greater discrepancy between simulated and real-world behavior.

A. Scalability Analysis

The evolved robots described here demonstrate that automatically designing and fabricating a variety of different machines from the same base material is possible. However, the present implementation imposes a few design constraints. The serial topology of our system requires special care to avoid self-intersection during printing and robot actuation (a step handled automatically by our design software). Also, the choice of wire stiffness is a trade-off between the current printer's ability to bend the wire and the wire's ability to support the emerging structure of the robot; future systems with stiffer wire or alternative mechanisms (for example a fluid-bath or micro-gravity environment) to support the robot as it is being folded could allow much longer robots to be fabricated. Though the individual morphology of the robot being fabricated dictates the size of the self-supporting structure, we can bound the problem by performing a worst-case analysis based on a cantilevered configuration. Equation 2 relates the maximum cantilevered length (L) to robot design-length (L^*), wire density (ρ) and diameter (d), motor mass (m_m), spacing between motors (S), and wire yield stress (σ_y):

$$\frac{1}{8}\rho\pi g d^2 L^2 + m_m g \sum_{n=1}^{\lfloor L^*/S \rfloor} (L - nS) - \frac{1}{6}\sigma_y d^3 = 0. \qquad (2)$$

Solving for L reveals that the longest cantilevered portion that can be manufactured with the current configuration is 1.39 meters with one motor module (54 folds with a 25 mm fold spacing). Robots with a longer overall-length are possible, provided that they are folded to remain within this maximum cantilevered radius of the bending head.

Extending this analysis to examine the space of possible materials reveals that those with a higher specific stiffness such as stainless steel or NiTinol wire (Fig. 5) could produce robots with twice the cantilevered length of those fabricated with aluminum wire because these relatively lighter and stiffer designs would allow longer sequences of folds to self-support. Regardless of the length of the robot enabled by a particular material choice, because the CPPN representation is generative, it already supports seamlessly scaling to robot designs with higher complexities (see Fig. 6).

The cold-working involved in bending aluminum wire eventually results in its failure through brittle fracture. This limits the number of recycling steps to between 10^1 to 10^2 cycles,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LRA.2017.2716418, IEEE Robotics and Automation Letters

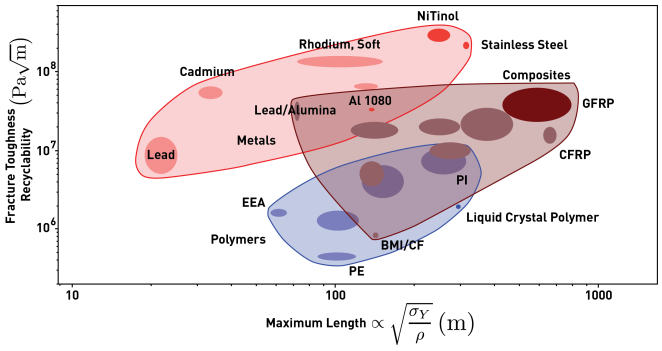

Fig. 5: **Analysis of Candidate Feed Materials**. The maximum length of a design depends on the yield strength of the feed material; higher yield strength enables longer segments and more complex designs. The failure mode of recycled material is brittle fracture due to work hardening. Therefore, fracture toughness is a proxy for the recyclability of a material.

before the material shatters [42]. The alternative materials shown in Fig. 5 also have fracture toughnesses that are superior to aluminum, making them better suited to the cold-working that is involved in recycling.

VI. Discussion

Robots are currently designed and fabricated manually, leading to high costs and making them time-consuming to produce or adapt to novel scenarios. To address this issue, and enable robots that are simultaneously tailored to an application and inexpensive, roboticists have begun to break from the reliance on manual design and fabrication by using modular design approaches and automated fabrication methods. Recent work with rigid [43]–[45] and soft robots [46], [47] has employed a combination of interactive design based on manual composition of modules from a library pre-populated by expert-designers, and a subsequent optimization step to refine these mechanisms based on the application's objectives. In all cases, multiple stages of manual fabrication and assembly were involved to implement the designs. New additive manufacturing techniques have been developed to automatically fabricate complete assembly-free robot mechanisms [1], however, substantial human decision-making was required during the design phase of these systems.

In contrast, by leveraging ideas from natural assembly processes we have demonstrated that automatically designing *and* fabricating a variety of different robots is possible. Although the robots shown here have modest functionality, the process and modules used are readily scaled to permit larger or more complex robots. For example, the wireless communication links between the motor modules are bidirectional and transmit sensor data as well as motor control commands. If necessary, a module with more internal volume could accommodate a more powerful processor or additional sensors. Thicker and stiffer wire (selected from Fig. 5) combined with more powerful motor modules would allow stronger robots. Provided that the total cantilevered length of the robot adheres to eq. 2, there is no limit on the linear length of the robots that can be bent; robots more complex than those shown

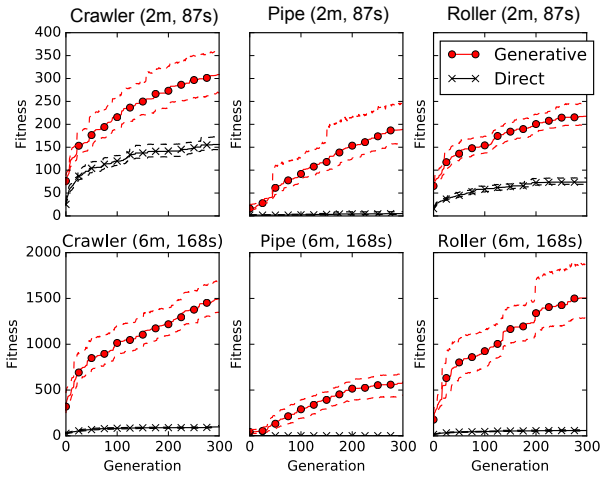

Fig. 6: **Generative vs. Direct Representation.** We compare the generative CPPN with a direct encoding, in which every fold is described by a separate parameter in the genotypic description. Both methods are tasked with evolving robots with two motors and 87 segments (**top**) as well as larger robots with six motors and 168 segments (**bottom**) for three navigation tasks. Shown are mean F_1 values (eq. 1) over 20 evolutionary runs together with their 95% bootstrapped confidence intervals. The CPPN encoding finds high performing robots with varying complexity while the direct encoding often struggles to find designs that simultaneously satisfy all fitness criteria.

could be capable of grasping and manipulating objects in the environment by rotating adjacent body segments relative to each other. Additionally, a similar approach could allow multiple robot chains to connect after the printing process: simply bringing the robots into contact with each other would allow them to fold together and interlink, a concept inspired by protein bonding. The non-backdrivable actuators used in this system would provide zero-energy position-holding in the interlocked segments, allowing actuated hinges to be treated like permanent latches. This would allow multiple robot chains to combine, providing topological design flexibility: rather than simple chains, combined structures with branching (arm-like) features would be achievable.

Designing and fabricating specialized robots on-demand will allow them to be customized for each application, rather than using more expensive machines that are exhaustively designed to be general-purpose. This advance could enable robots to be rapidly adapted to disaster scenarios or high-risk environments, in which the challenges are not known a priori; the robot deployment might take a phased approach in which observer robots assess the scenario and are then followed by customized robots produced on-demand to address the specific need (e.g. longer legs to surmount an obstacle; a gripper whose shape is customized to reach and grasp an otherwise inaccessible object). Similarly, the ability to adapt to unknown situations could be valuable in inaccessible or remote areas, including space exploration. This approach points in a new direction, toward *expendable robotics*, in which customized robots are rapidly fabricated on-demand, consumed by their application, and are then recycled.

REFERENCES

[1] R. MacCurdy, R. Katzschmann, Y. Kim, and D. Rus, "Printable hydraulics: A method for fabricating robots by 3D co-printing solids and liquids," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 3878–3885.

[2] R. MacCurdy, A. McNicoll, and H. Lipson, "Bitblox: A printable digital material for electromechanical machines," *International Journal of Robotics Research*, vol. 33, no. 10, pp. 1342–1360, 2014.

[3] S. B. Walker and J. A. Lewis, "Reactive silver inks for patterning high-conductivity features at mild temperatures," *Journal of the American Chemical Society*, vol. 134, no. 3, pp. 1419–1421, 2012.

[4] S.-Y. Wu, C. Yang, W. Hsu, and L. Lin, "3d-printed microelectronics for integrated circuitry and passive wireless sensors," *Microsystems & Nanoengineering*, vol. 1, 2015.

[5] H. Lipson and J. B. Pollack, "Automatic design and manufacture of robotic lifeforms," *Nature*, vol. 406, no. 6799, pp. 974–978, 2000.

[6] A. M. Mehta, J. DelPreto, B. Shaya, and D. Rus, "Cogeneration of mechanical, electrical, and software designs for printable robots from structural specifications," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 2892–2897.

[7] S. Coros, B. Thomaszewski, G. Noris, S. Sueda, M. Forberg, R. W. Sumner, W. Matusik, and B. Bickel, "Computational design of mechanical characters," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 4, p. 83, 2013.

[8] S. Felton, M. Tolley, E. Demaine, D. Rus, and R. Wood, "A method for building self-folding machines," *Science*, vol. 345, no. 6197, pp. 644–646, 2014.

[9] S. Griffith, "Growing machines," Ph.D. dissertation, Massachusetts Institute of Technology, 2004.

[10] K. C. Cheung, E. D. Demaine, J. R. Bachrach, and S. Griffith, "Programmable assembly with universally foldable strings (moteins)," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 718–729, Aug 2011.

[11] A. Knaian, K. Cheung, M. Lobovsky, A. Oines, P. Schmidt-Neilsen, and N. Gershenfeld, "The Milli-Motein: A self-folding chain of programmable matter with a one centimeter module pitch," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 1447–1453.

[12] E. Hawkes, B. An, N. M. Benbernou, H. Tanaka, S. Kim, E. D. Demaine, D. Rus, and R. J. Wood, "Programmable matter by folding," *Proceedings of the National Academy of Sciences*, vol. 107, no. 28, pp. 12 441–12 445, 2010.

[13] S. Miyashita, S. Guitron, M. Ludersdorfer, C. R. Sung, and D. Rus, "An untethered miniature origami robot that self-folds, walks, swims, and degrades," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1490–1496.

[14] Y. Mao, K. Yu, M. S. Isakov, J. Wu, M. L. Dunn, and H. J. Qi, "Sequential self-folding structures by 3d printed digital shape memory polymers," *Scientific reports*, vol. 5, 2015.

[15] D. Raviv, W. Zhao, C. McKnelly, A. Papadopoulou, A. Kadambi, B. Shi, S. Hirsch, D. Dikovsky, M. Zyracki, C. Olguin *et al.*, "Active printed materials for complex self-evolving deformations," *Scientific reports*, vol. 4, 2014.

[16] L. Wang, M. M. Plecnik, and R. S. Fearing, "Robotic folding of 2d and 3d structures from a ribbon," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 3655–3660.

[17] Z. Li and E. Diller, "Polymer filament–based in situ microrobot fabrication using magnetic guidance," *International Journal of Advanced Robotic Systems*, vol. 14, no. 1, p. 1729881416682707, 2016.

[18] K. Gilpin, A. Knaian, and D. Rus, "Robot pebbles: One centimeter modules for programmable matter through self-disassembly," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 2485–2492.

[19] A. Lyder, R. Garcia, and K. Stoy, "Mechanical design of Odin, an extendable heterogeneous deformable modular robot," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, Sept 2008, pp. 883–888.

[20] J. Neubert, A. P. Cantwell, S. Constantin, M. Kalontarov, D. Erickson, and H. Lipson, "A robotic module for stochastic fluidic assembly of 3d self-reconfiguring structures," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 2479–2484.

[21] J. Neubert and H. Lipson, "Soldercubes: a self-soldering self-reconfiguring modular robot system," *Autonomous Robots*, vol. 40, no. 1, pp. 139–158, 2016.

[22] N. Eckenstein and M. Yim, "Design, principles, and testing of a latching modular robot connector," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2014, pp. 2846–2851.

[23] L. Brodbeck, S. Hauser, and F. Iida, "Morphological evolution of physical robots through model-free phenotype development," *PloS one*, vol. 10, no. 6, p. e0128444, 2015.

[24] G. E. Fox, "Origin and evolution of the ribosome," *Cold Spring Harbor perspectives in biology*, vol. 2, no. 9, p. a003483, 2010.

[25] E. M. Sivertsson and L. S. Itzhaki, "Protein folding: When ribosomes pick the structure," *Nature chemistry*, vol. 6, no. 5, pp. 378–379, 2014.

[26] K. O. Stanley, "Compositional pattern producing networks: A novel abstraction of development," *Genetic Programming and Evolvable Machines Special Issue on Developmental Systems*, vol. 8, no. 2, pp. 131–162, 2007.

[27] J. Secretan, N. Beato, D. B. D'Ambrosio, A. Rodriguez, A. Campbell, J. T. Folsom-Kovarik, and K. O. Stanley, "Picbreeder: A case study in collaborative evolutionary exploration of design space," *Evolutionary Computation*, vol. 19, no. 3, pp. 373–403, 2011.

[28] J. Clune and H. Lipson, "Evolving 3D objects with a generative encoding inspired by developmental biology," *ACM SIGEVOlution*, vol. 5, no. 4, pp. 2–12, 2011.

[29] J. Gauci and K. O. Stanley, "Autonomous evolution of topographic regularities in artificial neural networks," *Neural Comput.*, vol. 22, pp. 1860–1898, 2010.

[30] S. Risi, D. Cellucci, and H. Lipson, "Ribosomal robots: Evolved designs inspired by protein folding," in *Proc. of the 15th annual conference on Genetic and evolutionary computation*. ACM, 2013, pp. 263–270.

[31] J. E. Auerbach and J. C. Bongard, "Environmental influence on the evolution of morphological complexity in machines," *PLoS Comput Biol*, vol. 10, no. 1, pp. 1–17, 01 2014.

[32] G. S. Hornby, H. Lipson, and J. B. Pollack, "Evolution of generative design systems for modular physical robots," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, vol. 4, 2001, pp. 4146–4151 vol.4.

[33] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[34] A. E. Eiben and J. Smith, "From evolutionary computation to the evolution of things," *Nature*, vol. 521, no. 7553, pp. 476–482, 2015.

[35] J. C. Bongard, "Evolutionary robotics," *Communications of the ACM*, vol. 56, no. 8, pp. 74–83, 2013.

[36] S. Lloyd, "Computational capacity of the universe," *Physical Review Letters*, vol. 88, no. 23, p. 237901, 2002.

[37] K. O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evolutionary Computation*, vol. 10, no. 2, pp. 99–127, 2002. [Online]. Available: http://nn.cs.utexas.edu/?stanley:ec02

[38] ——, "Competitive coevolution through evolutionary complexification," vol. 21, pp. 63–100, 2004.

[39] J. Lehman and K. O. Stanley, "Abandoning objectives: Evolution through the search for novelty alone," *Evolutionary computation*, vol. 19, no. 2, pp. 189–223, 2011.

[40] ——, "Evolving a diversity of virtual creatures through novelty search and local competition," in *Proceedings of the 13th annual conference on Genetic and evolutionary computation*. ACM, 2011, pp. 211–218.

[41] A. Cully, J. Clune, D. Tarapore, and J.-B. Mouret, "Robots that can adapt like animals," *Nature*, vol. 521, no. 7553, pp. 503–507, 2015.

[42] J. MARIN, "Cumulative damage and effect of mean strain in low-cycle fatigue of a 2024-t351 aluminum alloy," *Journal of Basic Engineering*, vol. 8, pp. 0–1, 1966.

[43] A. Schulz, C. Sung, A. Spielberg, W. Zhao, Y. Cheng, A. Mehta, E. Grinspun, D. Rus, and W. Matusik, "Interactive robogami: Data-driven design for 3d print and fold robots with ground locomotion," in *SIGGRAPH 2015: Studio*. New York, NY, USA: ACM, 2015.

[44] C. Sung and D. Rus, "Foldable joints for foldable robots," *Journal of Mechanisms and Robotics*, vol. 7, no. 2, p. 021012, 2015.

[45] A. Mehta, J. DelPreto, and D. Rus, "Integrated codesign of printable robots," *Journal of Mechanisms and Robotics*, vol. 7, no. 2, 2015.

[46] F. Connolly, C. J. Walsh, and K. Bertoldi, "Automatic design of fiber-reinforced soft actuators for trajectory matching," *Proceedings of the National Academy of Sciences*, 2016.

[47] M. Wehner, R. L. Truby, D. J. Fitzgerald, B. Mosadegh, G. M. Whitesides, J. A. Lewis, and R. J. Wood, "An integrated design and fabrication strategy for entirely soft, autonomous robots," *Nature*, vol. 536, no. 7617, pp. 451–455, 2016.