Robotics: Science and Systems 2024
Delft, Netherlands, July 15-July 19, 2024

Evolution and learning in differentiable robots

Luke Strgar, David Matthews, Tyler Hummer, Sam Kriegman
Northwestern University

***Abstract*—The automatic design of robots has existed for 30 years but has been constricted by serial non-differentiable design evaluations, premature convergence to simple bodies or clumsy behaviors, and a lack of sim2real transfer to physical machines. Thus, here we employ massively-parallel differentiable simulations to rapidly and simultaneously optimize individual neural control of behavior across a large population of candidate body plans and return a fitness score for each design based on the performance of its fully optimized behavior. Non-differentiable changes to the mechanical structure of each robot in the population—mutations that rearrange, combine, add, or remove body parts—were applied by a genetic algorithm in an outer loop of search, generating a continuous flow of novel morphologies with highly-coordinated and graceful behaviors honed by gradient descent. This enabled the exploration of several orders-of-magnitude more designs than all previous methods, despite the fact that robots here have the potential to be much more complex, in terms of number of independent motors, than those in prior studies. We found that evolution reliably produces "increasingly differentiable" robots: body plans that smooth the loss landscape in which learning operates and thereby provide better training paths toward performant behaviors. Finally, one of the highly differentiable morphologies discovered in simulation was realized as a physical robot and shown to retain its optimized behavior. This provides a cyberphysical platform to investigate the relationship between evolution and learning in biological systems and broadens our understanding of how a robot's physical structure can influence the ability to train policies for it. Videos and code at https://sites.google.com/view/eldir.**

I. Introduction

Robots are usually trained automatically but designed manually through extensive trial and error. The ability to automatically generate and test large numbers of design variants could allow for new approaches to robotics that venture into previously unknown parts of design space, revealing forms and functions that are well suited for their environment but that may not be immediately obvious to human engineers. However, despite three decades of research (Table I), the automatic design of robots remains in its infancy, and surprisingly few designs have been explored in simulations let alone reality.

Co-optimizing a robot's physical layout and its control policy is challenging due to the bilevel and combinatorial nature of the search problem. Many have attempted to collapse this hierarchical problem to a single level, and more recently, to transform the discrete physical structure of a robot into a more continuous and thereby more differentiable form [45]. However, this has required imposing heavy constraints on either the robot's body or its brain, and usually results in premature morphological convergence very early in optimization, effectively transforming the design problem into one of policy training for a randomly generated and morphologically simple agent [11].

Others [62, 66, 18, 3] have respected the natural bilevel organization of the problem, nesting the controller optimization task inside of the morphology optimization task as a constraint. Changes in robot design, like changes in animal design, occurred on a much slower (evolutionary) timescale than that of changes to neural control (learning). However, the exorbitant computational cost of the inner optimization task—policy training using non-differentiable physics simulators to evaluate agent designs, and simulating each one forward one body part at a time—shackled the outer optimization task to a vanishingly small corner of design space: Only a few thousand designs could be explored, and those designs were relatively simple, possessing very few degrees of freedom, and even fewer motors. Despite these simplifying assumptions, no such robots have been constructed in the material world.

Differentiable simulation [20, 15, 25, 26] has recently made possible the efficient gradient-based optimization of robots, opening new paths to body-brain co-optimization. However, there are only four cases reported in the literature [44, 45, 65, 13] in which the physical aspects of a robot's design were allowed to change during gradient-based optimization, and none tackled the brain-body problem head on. Either the body was heavily constrained (i.e. it was interpolated between predefined basis shapes [44]) or there was no brain (i.e. actuation patterns were predefined and did not incorporate sensory feedback [45, 13, 65]). All four methods collapsed the problem to a single level and greedily optimized the robot's body plan by locally descending whatever design gradient happened to be nearby its random initial conditions, precluding exploration of the wider design space. Moreover, they all assumed simple task environments with perfectly flat terrain. And though the optimized design from Matthews et al. [45] was realized as a physical robot, it is non-trivial to generalize gradient-based morphological search to arbitrary hardware constraints, such as those imposed by the relatively large, discrete and functional building blocks (motors, sensors, battery packs, wires, gears, tendons, wheels, etc.) that are common in embodied machines because they simplify design, manufacture, and repair, but are not naturally amenable to continuous modeling.

The automatic design of robots began with a simple genetic algorithm, and, until recently, stochastic gradient-free population-based methods were the only viable option. Though this is no longer the case, evolution remains an attractive complement to gradient descent for robot design, particularly in the outer morphological level of search. Indeed, evolutionary algorithms are embarrassingly parallel, simple to implement with arbitrary constraints and multiple objectives, and can explore non-differentiable, rugged, sparse, and deceptive loss

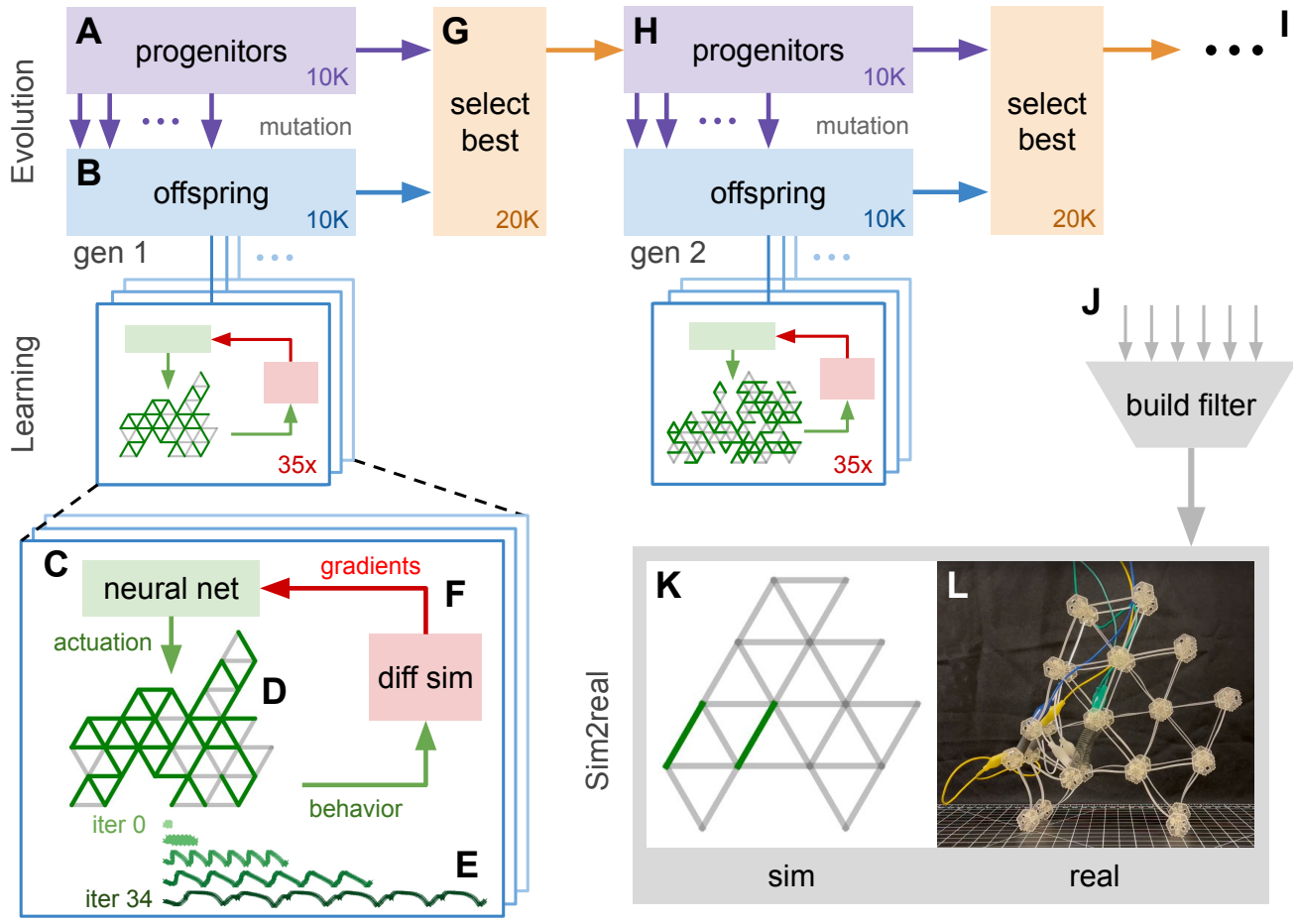

Fig. 1. **Evolution and learning.** An initially random population of 10K progenitor robots (**A**) produce 10K offspring (**B**) through random morphological mutations and/or crossover, temporarily doubling the size of population to 20K robots. The 10K offspring are then simultaneously trained in parallel differentiable simulations as follows. Each robot has its own proprioceptive neural network (**C**) that coordinates the actuation of its motors (**D**) through differentiable simulation to produce behavior (**E**), yielding a performance score based on the net displacement of the robot in the desired direction of travel (into the right hand side of the page). Gradients are then propagated backward in time through the simulated behavior (**F**) and used to update the neural net's initially random synaptic weights such as to improve its performance. This process is repeated for 34 additional iterations of gradient descent (35 total gradient descent steps), after which the robot's fitness was taken to be the best performance score it achieved during training. Finally, selection (**G**) reduces the population back to 10K robots by deleting the worst performing robots. Evolution then proceeds to the next generation (**H**) and this process of design variation, parallel differentiable training, and selection is repeated 998 times for a total of 1000 generations of evolution (**I**). A build filter (**J**) was then used to identify the most manufacturable designs discovered in simulation; e.g. the simulated design in **K** which was printed (**L**) and fitted with shape memory alloy springs that can be repeatedly energized and cooled to generate forward locomotion.

landscapes without becoming trapped in local optima. They can also shine unique light on biological evolution by generating new hypotheses and testing those that would be otherwise difficult or impossible to test in vitro or vivo [40, 33]. However, the full benefits of evolutionary approaches to robot design, whatever they may be, have not been realized due to lack of scale.

One such potential benefit could be the ability to simultaneously optimize many robots in the same population with differing physical forms and control policies, thus realizing the desired behavior in unique ways (e.g.) along a Pareto front of competing objectives, or evenly distributed within a projection of design space. Indeed, maintaining a diverse population of suboptimal designs can in some cases be more useful than the single optimal design for a set of explicit objectives because no simulation is perfect and some desired behaviors and structures are less realizable than others (s.v. the simulation-reality gap [27]) and because it is often challenging to specify a full range of desired and undesired behaviors a priori (s.v. the alignment problem [6]). However, the largest population used to evolve robots has been limited to 10^3 individuals [39], and usually contains many fewer (Table I).

For several decades, the maximum number of designs explored across an entire run of evolution has been on the order of 10^5. However, it is unclear how many unique body plans were actually visited by brain-body co-optimization as search was typically collapsed to a single level—a single timescale of adaption—in which any design revision could have been applied only to the controller and left the morphology unchanged. In such cases, the total number of designs explored was computed in Table I as the population size multiplied by the number of generations of evolution. This is a rather

TABLE I
SUMMARY OF NONPARAMETRIC MORPHOLOGICAL EXPLORATION WITHIN A SINGLE RUN OF AUTOMATIC ROBOT DESIGN.[1]

Author/citation	*Year*	*Material*	*Adaptive timescales*	*Independent motors*	*Pop size*	*Designs explored*	*Realized physically*
Sims [57]	1994	Rigid	Single	∼12	300	30K	No
Ventrella [61]	1994	Rigid	Single	∼12	100	1K	No
Lipson and Pollack [43]	2000	Rigid	Single	∼12	200	100K	**Yes**
Komosiński and Rotaru-Varga [30]	2001	Soft	Single	∼5	200	60K	No
Bongard and Pfeifer [4]	2003	Rigid	Single	49	200	40K	No
Hornby et al. [24]	2003	Rigid	Single	**349**	100	50K	**Yes**
Miconi and Channon [47]	2006	Rigid	Single	∼12	200	100K	No
Chaumont et al. [8]	2007	Rigid	Single	∼3	300	30K	No
Hiller and Lipson [21]	2010	Soft	Single	2	20	25K	No
Krčah [32]	2010	Rigid	Single	∼12	300	45K	No
Auerbach and Bongard [1]	2011	Rigid	Single	∼20	150	75K	No
Lehman and Stanley [39]	2011	Rigid	Single	∼12	**1000**	500K	No
Hiller and Lipson [22]	2012	Soft	Single	1	50	125K	**Yes**
Lessin et al. [41]	2013	Rigid	Single	∼12	200	400K	No
Cheney et al. [9]	2013	Soft	Single	2	30	30K	No
Cheney et al. [10]	2014	Soft	Single	1	30	30K	No
Rieffel et al. [55]	2014	Soft	Single	1	20	10K	No
Auerbach and Bongard [2]	2014	Rigid	Single	2	150	75K	No
Brodbeck et al. [5]	2015	Rigid	Single	4	10	100	**Yes**
Joachimczak et al. [28]	2016	Soft	Single	2	300	**600K**	No
Cellucci et al. [7]	2017	Soft	Single	2	100	30K	No
Cheney et al. [12]	2018	Soft	Single	1	50	250K	No
Kriegman et al. [34]	2018	Soft	Single	1	25	125K	No
Corucci et al. [14]	2018	Soft	Single	1	10	90K	No
Wang et al. [62]	2019	Rigid	**Bilevel**	∼5	64	13K	No
Pathak et al. [53]	2019	Rigid	Single	5	N/A	1K	No
Kriegman et al. [35]	2020	Bio	Single	1	50	50K	**Yes**
Kriegman et al. [36]	2020	Soft	Single	1	N/A	7K	**Yes**
Veenstra and Glette [60]	2020	Rigid	Single	19	100	50K	No
Miras et al. [48]	2020	Rigid	Single	14	100	10K	No
Zhao et al. [66]	2021	Rigid	**Bilevel**	∼20	1	2K	No
Kriegman et al. [37]	2021	Bio	Single	1	16	8K	**Yes**
Kriegman et al. [38]	2021	Soft	Single	2	16	5K	**Yes**
Gupta et al. [18]	2021	Rigid	**Bilevel**	10	576	4K	No
Moreno and Faiña [49]	2021	Rigid	Single	∼10	40	25K	**Yes**
Hejna III et al. [19]	2021	Rigid	Single	9	24	1K	No
Bhatia et al. [3]	2021	Soft	**Bilevel**	25	25	1K	No
Medvet et al. [46]	2021	Soft	Single	25	100	30K	No
Ma et al. [44]	2021	Soft	Single	2	1	25	No
Yuan et al. [64]	2021	Rigid	Single	∼14	N/A	1K	No
Schaff et al. [56]	2022	Soft	Single	8	N/A	50K	**Yes**
van Diepen and Shea [59]	2022	Soft	Single	8	1	100	No
Norstein et al. [51]	2023	Rigid	Single	20	64	8K	No
Pigozzi et al. [54]	2023	Soft	Single	100	100	30K	No
Matthews et al. [45]	2023	Soft	Single	1	1	10	**Yes**
Yuhn et al. [65]	2023	Soft	Single	4	1	1K	No
Cochevelou et al. [13]	2023	Soft	Single	2	1	200	No
Li et al. [42]	2024	Soft	Single	2	N/A	320K	No
This study	2024	Soft	**Bilevel**	**453**	**10K**	**10M**	**Yes**

generous upper bound. By this count it was Joachimczak et al. [28] that reported the largest prior exploration, with 600K design revisions in a single run of optimization, but the winning body plan "emerge[d] in the first few hundreds of generations" and the remainder of the run was spent training policies for it.

Design exploration has also been limited by an overall lack of morphological complexity. Two decades ago, Hornby et al. [24] used a grammar-based encoding to recursively generate kinematic chains with up to 349 independently oscillating joints. These joints, however, were blindly actuated in an open loop without incorporating sensory information. Since then, no automatically designed robot has possessed over a hundred independent motors. When allowed to incorporate larger numbers of actuators [10, 12], locomotion was coordinated by a global pacemaker which propagated a sinusoidal wave of excitation down the robot's entire body, resulting in simple periodic gaits.

Here we evolve populations of tens of thousands of morphologically complex robots in parallel differentiable simulations, rapidly acquiring a suitable closed-loop control policy for each design using gradient descent (Fig. 1). Combining global (evolutionary) morphological search with local (gradient based) policy training obviates the drastic premature design convergence suffered by previous work and generates

[1]It is important to note that exploring a large subset of design space was not always the goal of prior work in Table I. Indeed, sometimes the goal was to minimize exploration in rapid pursuit of a locally optimal design [45].

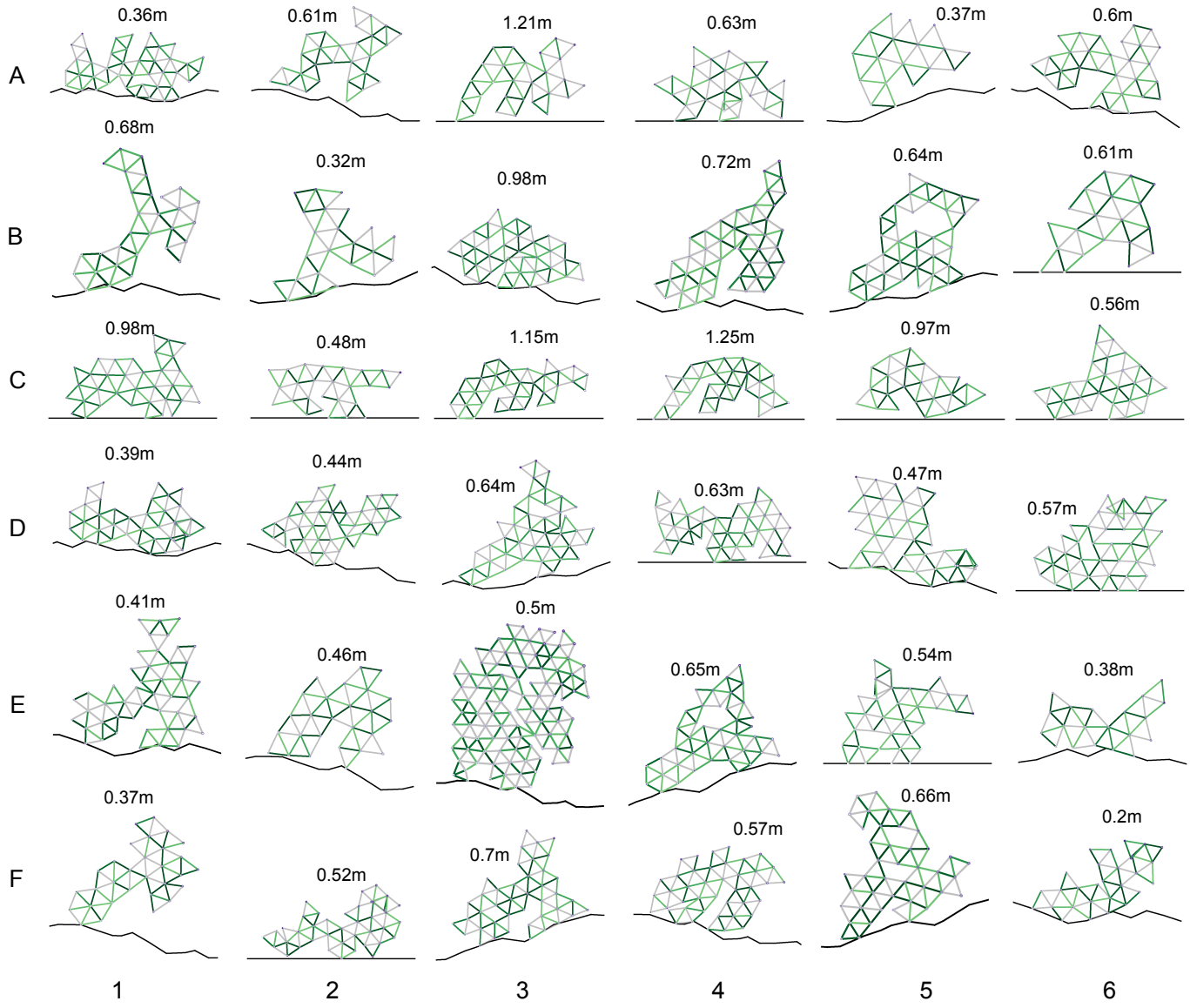

Fig. 2. **A diversity of body shapes** were discovered by evolution, each with its own unique internal distribution of active (green) and passive (gray) springs. The optimized performance (displacement in the desired direction of travel) achieved by gradient based learning is denoted above each design in meters (m). On rugged terrains bipedal body plans often emerged (A2, A6, B1, B4, B5, D3, E4, F3, F5) whereas on flat terrain robots often evolved three or four legs (A3, C1, C3, C4). Occasionally, swinging limbs that did not make contact with the ground were used to generate forward momentum (A4, B1, B4, E5). Videos and code can be found at **https://sites.google.com/view/eldir**.

a steady stream of complex body plans with finely tuned sensorimotor coordination and agile behaviors along complex terrains (Fig. 2). This allowed for exploration of several orders-of-magnitude more robot designs than any other method reported to date. Moreover, because the simulations themselves are parallelized across body parts, robots here may contain much greater motoric complexity (i.e. many more independent motors) than any previously evolved robots.

II. Methods

This section describes the space of possible robot bodies (Sect. II-A), their neural controllers (Sect. II-B), the differentiable simulations (Sect. II-C) in which they were evaluated and trained (Sect. II-D), how different environments were modeled (Sect. II-E and II-F), the genetic algorithm that traverses design space (Sects. II-G, II-H and II-I), and the sim2real transfer of evolved robots (Sect. II-J).

A. Morphology.

Each robot was composed of S springs connecting M masses on a triangular lattice. The lattice was of size A triangles wide and B triangles tall such that the height of the lattice was approximately equal to its width. Specifically, triangles are assumed to be equilateral with side length of 2, thereby relating the height and width of a triangle by a factor

of $\sqrt{3}$. In order to achieve a roughly square aspect ratio in the design space (lattice), A and B were set such $A \approx B\sqrt{3}$.

Robot geometry (R_g). A rectangular binary mask of shape (B, A) determines the presence or absence of each possible triangle, and the largest connected component was taken to be the expressed morphology. Triangles sharing an edge were considered connected, and triangles sharing a vertex along the vertical axis are also constituted valid connections. In translating from the mask to a triangular lattice, we observed it was possible for all the edges of a cell to be present while its corresponding entry in the mask was zero. In such cases the mask was filled appropriately. Henceforth we denote this mask as R_g as it represents the robot's geometry.

Internal distribution of active and passive springs (R_s). A binary vector of length S, denoted R_s, determines which springs are capable of actuation (active) and which are not (passive) at every possible spring location on the lattice. The length of the vector is a result of the dimensions the grid, A and B, and was computed automatically by enumerating all masses on the grid and counting all possible springs connecting them. The rest lengths of active springs may vary $\pm 10\%$ and are controlled through time using a neural-network controller as described below in the following section.

B. Neural control.

A multi-layer perceptron (MLP; Fig. 3A) controls the actuation of each of the robot's active springs (Fig. 3B) and takes as input proprioceptive signatures across the robot's body (Fig. 3C,D) as well as a set of central pattern generators (CPGs; Fig. 3E).

The set of CPGs was adopted from Hu et al. [26] and consists of 10 sine waves with a fixed angular frequency, $\omega = 0.08\pi$, and equally spaced phase offsets. Multiple phase-shifted CPGs provide the controller an easy way to synchronize different patterns of actuation across the robot's body at distinct times throughout their gait cycle.

At each mass in the robot's body there are four mechanosensory neurons that measure proprioception—the velocity and the relative displacement of each mass, in the horizontal and vertical dimensions. Displacement and velocity are measured with respect to the robot's center of mass and the initial (motionless; zero velocity) state of the robot. Thus, the neural net of a robot with M_r masses has $10 + 4 * M_r$ inputs.

The MLP contains a single hidden layer with 32 units. The choice of using 32 hidden units was also adopted from Hu et al. [26]. Each hidden unit computes a linear combination of weights and input features plus a learnable bias before applying the $\tanh$ function to produce hidden layer activations.

Controller outputs are computed in a similar fashion with final values squashed on the interval [-1, 1]. Output activations are mapped one-to-one to active springs in the robot's body and used control spring actuation. Here, a value of -1 and 1 correspond to maximum contraction and expansion of a spring, respectively. Additional details of the mapping between neural controller outputs and physical simulation are provided below in Sect. II-C.

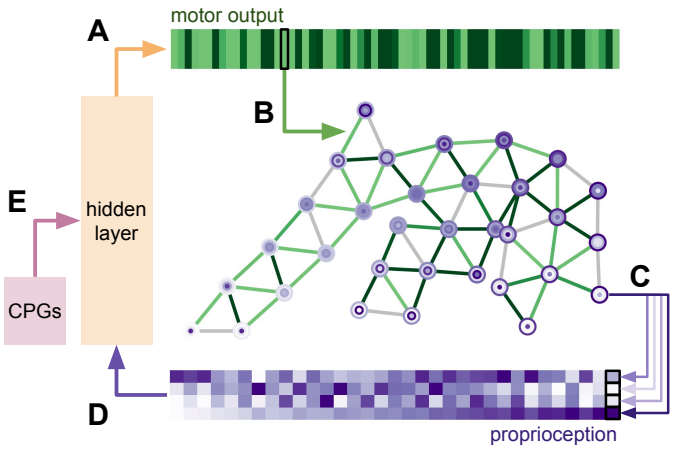

Fig. 3. **Neural control of behavior.** Each robot is controlled by a three-layer fully-connected neural network (**A**). At every time step, motor neurons output the rest lengths of each active spring in the robot's body (**B**). The sensory repercussions of these actions are captured by four proprioceptors at each of the robot's masses (concentric circles in **C**), which feed back into the nervous system (**D**) alongside central pattern generators (CPGs; **E**) closing the control loop. More precisely, there are 10 CPGs corresponding to 10 phase shifted sinusoidal waves; four proprioceptive channels track the vertical and horizontal velocity of each mass, as well as the vertical and horizontal displacement of each mass relative to the robot's center of mass, during behavior.

C. Simulation.

A Hookean spring-mass model was used to simulate robots. Masses are simulated using properties of mass and velocity as well as impulses resulting from spring actuation. Springs are simulated with properties including stiffness and length as well as constraints placed on maximum and minimum actuation strength. Actuated springs exert force on their end point masses by modulating their rest length to affect their strain and output force ($F = \Delta X * k$) for a given fixed spring stiffness k and variable displacement ΔX. In detail, each continuous neural controller (Sect. II-B) output value is used to compute a target rest length for the corresponding active spring, and Hooke's law is used to compute the force required to achieve the aforementioned target lengths. Subsequently, continuous valued impulses acting on endpoint masses of each spring are computed and used to simulate forward dynamics (Sect. II-C). Passive springs flexibly expand and contract under external forces (active springs, gravity, system dynamics) and exert impulses on their endpoint masses through tendency to preserve their fixed rest length.

We employed a GPU accelerated physics simulator (written in *Taichi* [26]) to simultaneously optimize neural controllers for all 10^4 robots in the population, evenly dividing them across 4 NVIDIA H100 SXM GPUs. At each time step, the actuation states and spring forces for each of the springs in each of the 10^4 robots was computed in parallel.

D. Learning.

The initially random synaptic weights of each robot's neural controller were drawn from a scaled Xavier Normal distribution [17] and then optimized for 35 iterations of gradient

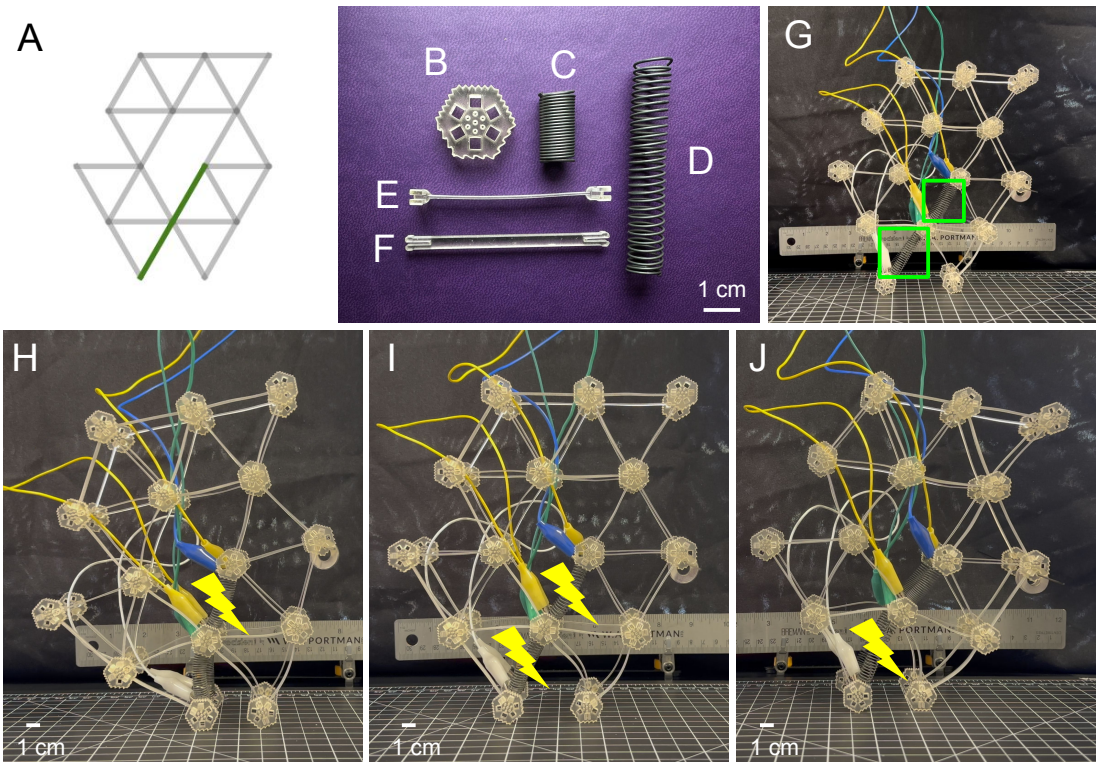

Fig. 4. **Manufacture.** Designs discovered in simulation (**A**) were assembled from 3D printed hexagonal masses (**B**), active two-way shape memory alloy springs (**C:** contracted; **D:** expanded), and passive springs (**E**). Two dimensional simulated designs were transferred to 3D physical robots by connecting two parallel planes of masses and springs with biplane connectors (**F**). (**G-J:**) The resultant robot has two active springs per planar face (four total; green boxes in G) that can be independently energized (lightning bolts in H-J) to deform the robot's body.

descent (Fig. 1C-F). This number of iterations was chosen because it proved sufficient in generating complex behaviors in our initial exploration of randomly generated designs. In each iteration, the robot was simulated forward for 1000 time steps with step size $dt = 0.004$, corresponding to an evaluation period of 4 seconds. Loss was computed as the net horizontal center-of-mass displacement of the robot in the "wrong" direction. That is, the negative of the net displacement of the robot's body in the desired direction of travel (i.e. into the right hand side of the page).

Gradients were computed using the automatic differentiation features provided by Taichi [26]. A distinct learning rate was computed for each neural controller as a constant multiple of the inverse root sum of squares of learnable parameter gradients [26]. Said parameters were then updated by applying a naive gradient descent step.

E. Terrain.

Robots were optimized for terrestrial locomotion across one of four distinct terrains. The first was a perfectly flat surface line and the other three were non-uniform, "rugged" terrains, which were modeled by a linear piece-wise function of angled ground surfaces. One of these four terrains was selected *a priori* and held fixed for the duration of an evolutionary run.

Rugged terrains were generated by a sequence of surface lines that were stitched together at their endpoints. Each line was defined by a slope and horizontal length, both drawn from uniform distributions on a manually fixed range of values. Component lines were sampled and appended to the ground until reaching a maximum horizontal length of 1.25 meters.

F. Ground contact model.

For flat ground (Sect. II-E) a time of impact (ToI) computation was used for mass-ground contacts. After advancing to the ToI, the horizontal and vertical component velocity of the mass were set to zero [26]. These choices result in an effective "no-slip" friction condition for robotic locomotion. Such a simple model may be justified for flat terrain; however, the no-slip condition is wildly inaccurate in other conditions, endowing simulated robots with gecko-like abilities to move across simulated vertical and inverted surfaces.

For rugged terrains the ground contact and friction model was therefore modified as follows. As in the no-slip condition, ToI was computed and the mass was advanced to the point of contact prior to applying friction. But here the friction force acting on a mass was set proportional to the minimum of the mass' component velocity magnitudes pointed normal and tangent to the ground. The mass' resulting velocity was then the original tangent component velocity less the friction force.

We found that using the minimum of the normal and tangent velocity magnitudes to estimate friction effectively captured desired contact behavior: where the normal component velocity was small, little to no friction force was applied, and where tangent component velocity was small (relative to the

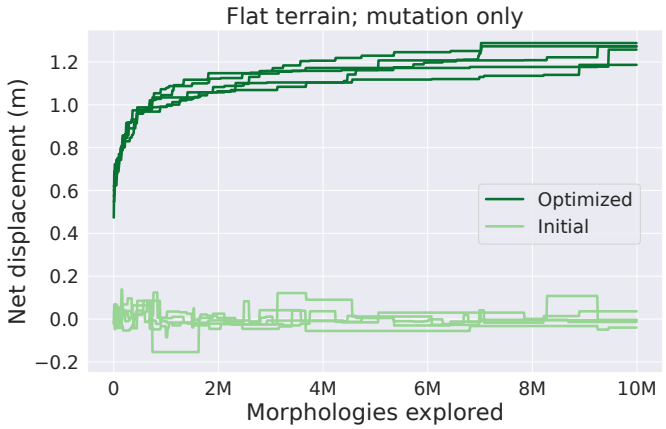

Fig. 5. **Increasingly differentiable robots.** The initially random behavior (light green) and the learned behavior (dark green) of the most performant design are plotted for five independent evolutionary trials (five pairs of light/dark lines) over flat terrain. Initial robot behavior (at the first iteration of gradient descent) produces little to no forward locomotion, whereas locomotive ability after learning continues to improve over evolutionary time. Each design was evaluated only once but may continue to survive and reproduce over many generations. Synaptic weights were not transmissible from parent to offspring.

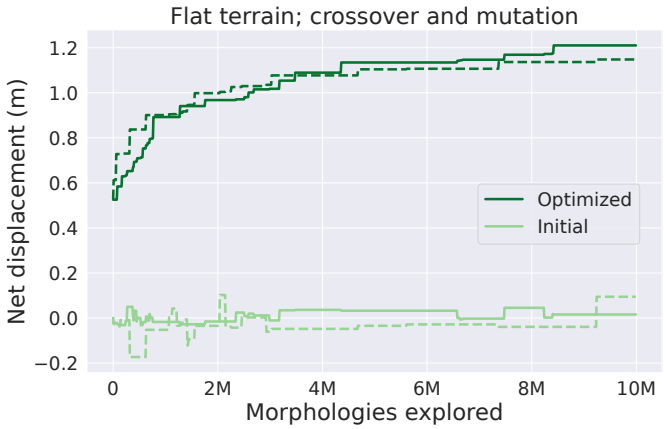

Fig. 6. **With crossover.** The initial (light green) and optimized (dark green) behavior of the most performant robot are plotted for two independent evolutionary trials in which offspring were produced through a combination of genetic recombination (crossover) and random mutation. The dashed and solid pairs of lines correspond to the "distinct" and "joint" masking approaches, respectively, used in crossover (described in Section II-H). As in Fig. 5, we see an increase in the differentiability of designs as measured by their improvement during gradient based learning. However, these fitness curves (dark green) rise at a slightly slower rate compared to the evolutionary trials that did not employ crossover.

normal component) the estimated friction force was never strong enough to flip the sign of the mass' tangent component velocity.

G. Mutation.

Before we describe the outer loop of morphological evolution, it is important to understand the primary variation operator: random mutation. The geometry and active spring state of a robot were randomly modified in its descendants through operations over the binary mask, R_g, and vector, R_s (defined in Sect. II-A). The geometry mask, R_g was mutated by randomly flipping bits with probability $\frac{1}{A*B}$ such that the expected number of flips is 1. Here, A and B denote the width and height of the triangular lattice (Sect. II-A). The largest connected component of the resulting mask was recomputed and compared with the original (parent) design. If no change had occurred the bit flipping procedure is repeated, doubling the flip probability each time. The spring vector R_s was randomly modified in a manner identical to that R_g; however, there was no requirement to observe a change since the geometry containing these springs was already guaranteed to be different. Additionally, neither R_s nor R_g was permitted to be empty (containing only zeros).

H. Crossover.

A second means of creating offspring robots involved combining distinct progenitor designs, typically referred to as crossover or genetic recombination. A pair of progenitor robots was randomly drawn from the population, and their geometry masks and spring vectors were fused to produce two offspring.

Spring vectors were combined by randomly selecting, with equal probability, the value from one of the two progenitors in each position of the vector and copying that value to the offspring's spring vector.

Two methods for combining morphology masks were explored. In both cases a bounding box for the largest connected component was found in each mask. The narrower of the two was then randomly shifted horizontally such that its width was entirely contained within the width of the wider bounding box, and an analogous operation was performed for the vertical axis. This ensured that a naive combination would include pieces of both progenitors. The first method—which we term *distinct masking*—randomly zeros an expected 35% of each morphology, unions the resulting masks and takes the largest connected component as the new morphology. The second method—which we term *joint masking*—first unions the masks then randomly zeros an expected 25% of the result and takes the largest connected component. In both cases each operation is performed twice to produce a pair of offspring. The genotype zero-masking probabilities were estimated empirically based on randomly-generated robot bodies.

I. Genetic algorithm.

Evolution was seeded with a set of 10K randomly generated progenitors, the control policies of which were trained using gradient based learning (Sect. II-D). Fitness was computed as the best behavior the robot achieved (absolute value of the lowest loss) across all 35 iterations of learning (Sect. II-D). That is, the largest net horizontal center-of-mass displacement of the robot during training. With the population thus initialized, evolution begins (Fig. 1A).

Every generation, each progenitor was mutated (as described in Sect. II-G) to produce an offspring (Fig. 1B), thereby doubling the population to 20K robots. In some experiments, crossover (detailed in Sect. II-H) was applied prior to mutation when producing offspring. In the experiments

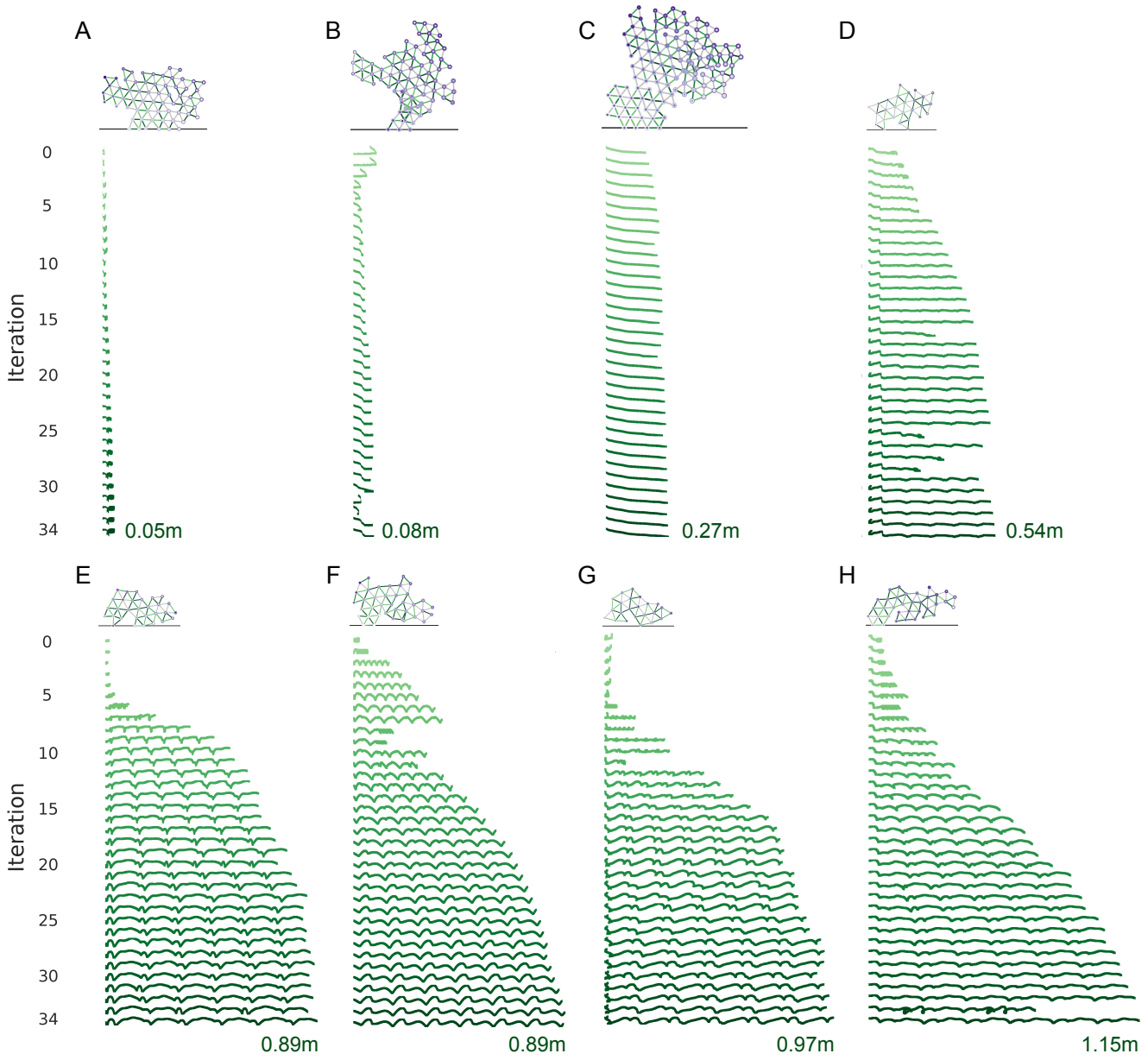

Fig. 7. **Learning to walk.** The behavioral trajectories of four robots that resisted learning (**A-D**) and four of the highly differentiable designs that evolved (**E-H**) are shown across 35 iterations of gradient descent (light to dark green CoM traces). Total horizontal displacement (fitness) is denoted to the right of (A-D) or below (E-H) the bottommost trace of each panel in meters (m). With randomly initialized synaptic weights (iteration 0), all robots are more or less sessile: they have not yet learned how to effectively control their bodies. (A:) Early in evolution, robots often demonstrated poor differentiability: they failed to learn to walk forward (into the right hand side of the page). (B,C:) Perverse instantiations of the objective function—"leaning towers"—evolved to be ever taller and tip over instead of walk. (D:) A moderately differentiable robot that later produced more effective offspring. (E-H:) Highly differentiable robots, on the other hand, show steady improvements in motility at each iteration of learning. In certain cases performance can abruptly deteriorate early (F,G) or late (H) in training before recovering for a net gain. The occasional deterioration in performance between iterations indicates opportunity for further development of the measure of robot differentiability.

in which crossover occurs, it does so with an 80% probability, leaving 20% of progenitors to produce offspring through mutation only. The employed 80% crossover likelihood was selected based on crossover probabilities found in the genetic programming literature, which typically range from 75%-95% [31]. In all experiments, offspring were optimized (as described in Sect. II-D) and evaluated in a manner identical to their progenitors (Fig. 1C-F). Offspring and progenitors were then sorted according to fitness (Fig. 1G) and the worst performing robots, whether progenitors or offspring, were deleted, reducing the population back to 10K robots: the next generation of progenitors (Fig. 1H). This process proceeds for 10^3 generations, resulting in the exploration of a total of 10^7 distinct robots.

We occasionally observed numerical instability during learning or in the resulting mass-spring system. When sorting

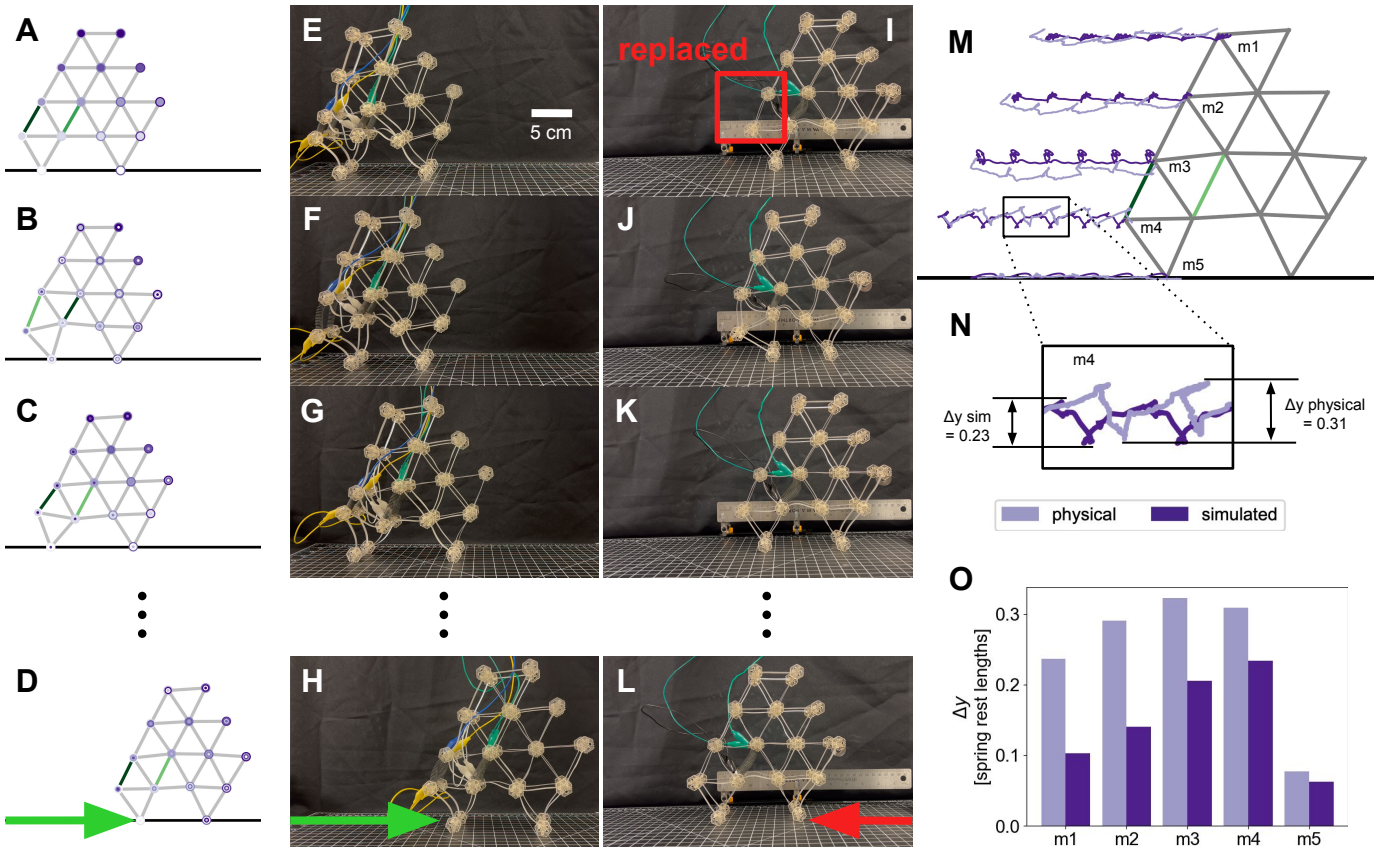

Fig. 8. **Sim2real.** A design discovered in simulation (**A-D**) locomotes over half a body length forward after 5 actuation cycles (D). A robot manufactured with the same design (**E-H**) achieved a similar displacement after 5 actuation cycles (H). To determine whether the robot's movement was a result of chance or due to the robot's evolved design, the design was altered by replacing one of its two motors with a passive spring (**I-L**). The physical instantiation of the evolved design moves in the desired direction of travel (green arrow in D and H), but, when one of its motors was replaced with a passive spring, it moved the "wrong way" (i.e. in reverse direction; red arrow in L). The physical trajectories (light purple traces) and the simulated trajectories (dark purple traces) of a set of masses near the actuating springs—those along the posterior edge of the design (m1-m5)—were tracked during behavior (**M**). Vertical range of motion (**N**) at these parts of the physical robot's body was relatively consistent with the simulator's predictions: the most movement occurred closest to the actuating springs (m3 and m4) and the least occurred at the mass in contact with the surface plane (**O**). Videos at **https://sites.google.com/view/eldir**.

by fitness value, robots with loss values from any training iteration equaling to NaN (not a number) or infinity were considered invalid and discarded. To account for unstable (chaotic) mass-spring system dynamics we also filtered out robots exhibiting large absolute difference in loss value from one iteration to the next. Empirically, we found large deviations to be indicative of a control policy resulting in unstable body mechanics, which could generate large, artificial body displacement scores. Across all conducted evolutionary runs this filtering criteria eliminated less than 0.83% of the robots evaluated and the average percentage of discarded robots in each run was 0.20% ($\pm$ 0.29% SD).

J. Construction.

To ensure evolved designs could be realized as physical robots, evolution was constrained in a second set of experiments to design robots within a 6×11 workspace and utilize at most two active springs (Fig. 4A).

Evolved robots were constructed by 3D printing three components: hexagonal masses (Fig. 4B), passive springs (Fig. 4E) and biplane connectors (Fig. 4F). Passive springs were fabricated with a length of 45mm and attached to the hexagonal masses to form the robot's triangular cells. Unlike simulated springs, passive physical springs were able to contract through bending but could not extend—relative to their fabricated length—under external forces. Passive physical springs exert forces on their endpoint masses through tendency toward their flattened rest length. The 2D simulated robot was extruded into 3D, and manufactured with two identical planar faces, which were offset from each-other using biplane connectors. Connectors were designed to be rigid so as to minimize motion of component masses out of their respective planes, and every mass having fewer than six in-plane spring connections was attached to one biplane connector.

Parts were printed on a Nexa3D XiP resin printer with x45 Clear resin (see Table IV for resin properties). After printing, parts were washed for two minutes while attached to the build plate in xCLEAN and subsequently one minute in IPA after being detached from the build plate. Parts were dried and then post cured for 25 minutes.

Active springs were constructed with Nitinol two-way shape memory alloy, which had a Martensite final transition temperature of 30C and an Austenite final transition temperature of 60C. In their high temperature state, the springs expand to be roughly 60mm long (Fig. 4D), and in their cool temperature state they contract to a length of 20mm (Fig. 4C). Active springs were actuated through joule heating (11V constant voltage) and placed within a stream of room temperature air to cool. This form of heat-based actuation is lightweight and provides both large deformation and high forces; however, it is much slower than is permitted in simulation, limiting the complexity of actuation sequences that the physical system can realize.

III. Results

A. The differentiability of robots.

Ten independent evolutionary runs were conducted in which robots were constrained on a 13×22 triangular lattice (up to 453 springs and 168 masses). For each run, the displacement of the most fit individual—evaluated after gradient-based learning—in each generation was plotted against the same robot's initial performance prior to learning (Figs. 5, 6 and 10). The increasing discrepancy between fitness before and after learning indicates the discovery of morphologies that were better suited to gradient-based control optimization in differentiable simulation. In other words, the differentiability of the best robots in the population increased over evolutionary time. The same evolutionary trend was observed throughout the population as indicated by the average fitness of the fully-trained robots increasing steadily without any appreciable increases in their initial fitness prior to learning (Fig. 11). Although synaptic weights were not transmitted from parents to their offspring, and there was no explicit selection pressure to increase the fitness of offspring prior to learning, robots could have evolved to be increasingly agnostic to synaptic weights [16] and thereby increasingly motile under random actuation patterns. An example of one such design that may exist in the search space but was not found by evolution is an asymmetrical wheel that rolls forward so long as there is sufficient internal actuation [33].

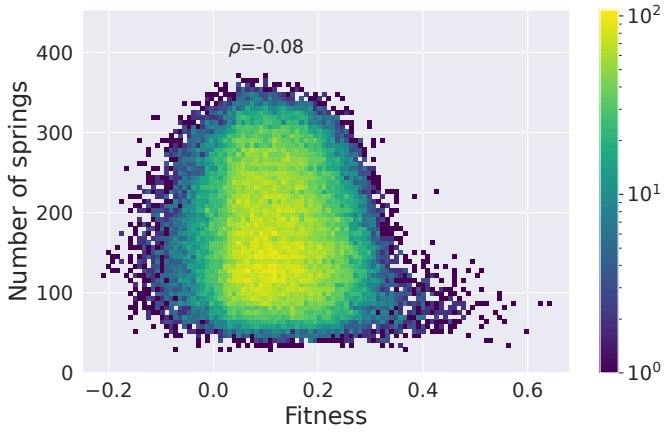

Fig. 9. **The relationship between size and fitness.** Body size (number of springs) and fitness was recorded for the 70K randomly-generated initial progenitor robots (gen 1) from the seven evolutionary conducted on flat ground terrain. The Spearman rank correlation coefficient was computed as $\rho = -0.08$, indicating little to no overall correlation between fitness and body size. However, the lower right tail of the distribution shows that the most fit randomly-generated designs posses relatively small bodies.

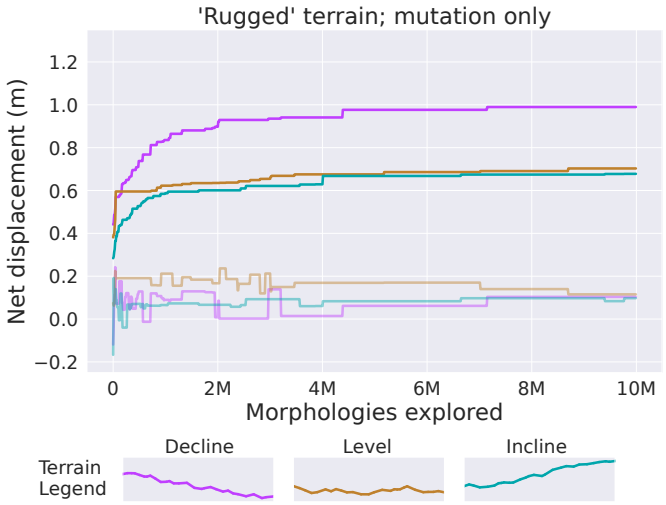

Fig. 10. **Evolution in more complex environments.** The initial (transparent) and optimized (opaque) behavior of the most performant design are plotted for three independent evolutionary trials over three distinct, rugged terrains. As in Figs. 5 and 6, the increasing divergence between initial and optimized fitness demonstrate the evolution of increasingly good learners, this time on more realistic (non-flat) terrains. Note that fitness values on rugged terrain are not directly comparable to those achieved on flat terrain due to different ground contact and friction models (described in Sect. II-F).

B. The effect of crossover.

To evaluate the potential for crossover to aid in morphological exploration, two of the ten independent trials included both crossover and mutation (Fig. 6).

As in the mutation-only experiments, evolution with crossover effectively selects for robot morphologies with increasingly higher locomotive potential (Fig. 6). However, in our small sample size (n=2), including crossover resulted in slightly worse performance among the most fit robots. This may indicate that either our chosen genotypic encoding or our chosen methods of merging genotypes during crossover could be improved, as discussed in Sect. IV.

C. Complex environments.

To investigate the evolution of learning in more complex environments—and under a more realistic friction model (Sect. II-F)—we conducted three additional mutation-only evolutionary runs over three distinct, rugged terrains (Sect. II-E). The shape of these surfaces can be seen in Figs. 2 and 10. We will refer to these environments by their net vertical tilt: "decline", "level", and "incline". As in the case of flat terrain (Figs. 5 and 6), evolution discovered body plans increasingly better at learning to move across these rugged terrains (Fig. 10). We emphasize that these robots were not

TABLE II
SIZE OF THE ROBOTS EVOLVED ON FLAT TERRAIN (TOTAL NUMBER OF SPRINGS).

	Gen 1	Gen 1000
All	176.15 (± 67.79 SD)	63.28 (± 13.85 SD)
Best	80.0 (± 10.61 SD)	55.0 (± 10.65 SD)
Largest	362.75 (± 2.05 SD)	84.0 (± 16.14 SD)

TABLE III
FRACTION OF SPRINGS THAT ARE ACTIVE WITHIN THE ROBOTS EVOLVED ON FLAT TERRAIN.

	Gen 1	Gen 1000
All	0.49 (± 0.04 SD)	0.72 (± 0.03 SD)
Best	0.56 (± 0.03 SD)	0.76 (± 0.03 SD)
Largest	0.50 (± 0.03 SD)	0.67 (± 0.03 SD)

augmented with any additional sensing capabilities relative to their flat terrain counterparts. Unsurprisingly, the fitnesses of best robots that move downhill ("decline") superseded those of robots that moved along "level" terrain, which in turn superseded the fitnesses of robots that climbed up a hill ("incline").

D. Morphology and fitness.

When evolved for locomotion across flat terrain (Sect. II-E) the size of the average, best, and largest robots in the population declined over the course of 1000 generations (Table II). This is due in part to a common local optimum of "leaning towers": tall, unstable bodies that oscillate until they tip over (Fig. 7B-C). Once discovered, it is easier for evolution to simply produce ever taller offspring that fall over ever farther than it is for evolution to create more stable offspring with revised geometries and motor distributions that enable gradient descent to learn a gait yielding bona fide locomotion with net displacement comparable to that of their leaning-tower parent.

Alternatively, robots with fewer motors may be easier to learn to control, creating selection pressure for small bodies. To test this theory, we considered the 10K randomly generated progenitors used to seed evolution in each of the seven independent trials on flat terrain. We ranked these 70K randomly generated robots according to their fitness and once again according to their size (number of springs). Spearman's rank correlation between the ranking of size and the ranking of fitness was close to zero ($\rho = -0.08$) and statistically significant ($p < 0.001$), indicating almost no correlation between fitness and robot size (Fig. 9). Nonetheless, even if smaller robots were only marginally more fit than large robots, the population should tend towards smaller sizes, which is what we observed.

This trend may also have been accelerated by the use of a constant mutation rate irrespective of robot size. The mutation rate could hypothetically render the discovery of adaptive mutations more difficult for larger robots where—relative to small robot—a higher number of mutations are required to introduce a new, adaptive morphological "feature" (e.g. a leg) at the appropriate scale.

While the total number of springs trended down, the fraction of active springs increased over the course of evolution (Table III), suggesting that greater motoric complexity (up to a point) is useful for learning effective behavior. Among the largest evolved robots in the population, the fraction of active springs is lower compared to the population aggregate and best, which may indicate that the employed method of neural control (Sect. II-B) was insufficient for controlling large bodies with many independent motors.

E. Sim2Real.

Unlike the simulated designs, the manufactured prototype operated with open-loop control. Active springs were manually actuated to imitate the actuation pattern and resulting gait of the closed-loop control policy observed in simulation following learning.

The evolved design selected for manufacture (Fig. 8A) moved forward 0.6 body lengths over the course of five simulated actuation cycles (Fig. 8A-D). The physical instantiation of this design (Fig. 8E) moved forward at a rate of 0.8 body lengths per five actuation cycles (Fig. 8E-H).

To determine if the physical robot's behavior could be attributed to the design and control discovered in simulation, we modified the robot by removing one of its active springs. The reduced robot completely lost the ability to locomote when actuated. To give the reduced robot a better chance at locomoting, additional mass (9g) was added to the anterior of the robot. When augmented in this way, the reduced robot regained the ability to locomote, albeit backwards (in the opposite direction), at rate of 0.4 body lengths over five actuation cycles (Fig. 8I-L).

To gain further insight into the design's behavioral transfer, the trajectories of the masses along the posterior edge were

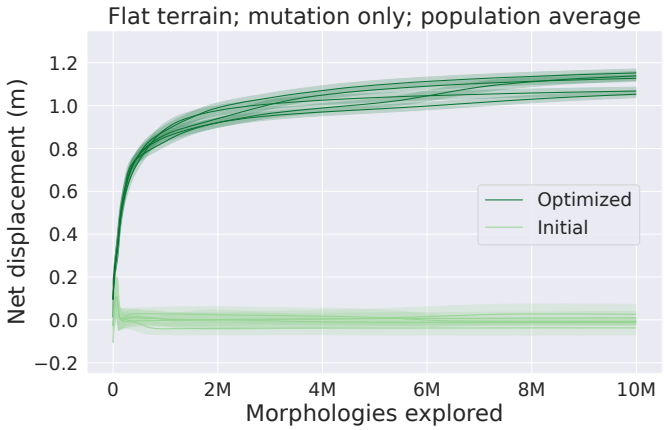

Fig. 11. **Mean fitness across the population.** The average initial fitness (light green) and average optimized fitness (dark green) across all 10K designs in the current population is shown at every generation (±1 S.D.) for the five mutation-only evolutionary trials conducted on flat terrain. Unlike Figs. 5, 6, and 10, where the fitness of the most performant design is shown at every generation, here it is demonstrated the population at large contains increasingly good learners and is increasingly fit overall.

TABLE IV
MATERIAL PROPERTIES OF THE NEXA3D X45 CLEAR RESIN USED TO PRINT THE STRUCTURE OF THE PHYSICAL ROBOTS.

ASTM code	*Property*	*Value*
D638	Tensile Modulus	1600 MPa
D638	Ultimate Tensile Strength	52 MPa
D638	Tensile Elongation at Break	12%
D790	Flex Modulus	2100 MPa
D790	Flex Strength	95 MPa
D256	Notched Izod	19 J/m
D570	Water Absorption	6%
D2240	Hardness Shore D	85

tracked and plotted for both the simulated and physical bodies (Fig. 8M). The vertical range of motion, Δy, of these mass trajectories (Fig. 8N), was extracted and plotted for each of the masses (Fig. 8O). The resulting range of motion for the physical system was relatively consistent with the simulation: most motion occurred at the mass locations closest to the active springs, and the least motion occurred at the mass in contact with the surface plane.

IV. DISCUSSION

The astonishing variety and complexity of animals that inspires roboticists today was designed automatically by a simple yet massively-parallel genetic algorithm. In this paper, we demonstrated that a similar evolution may be possible in robots if the bottleneck of serial non-differentiable evaluations is removed and replaced by a much wider, differentiable channel. However, there were several limitations of our initial approach that may be overcome by future work. What follows is a list of "problems" (they may, after all, turn out to be false problems) and potential solutions to each one. We hope that outlining problem-solution pairs in this way inspires others to engage creatively with the brain-body problem in robots and ultimately banish the technological obstacles that stand in the way of autonomous robots with morphological complexity and behavioral competency rivaling that of their living counterparts.

A. Limitations of differentiable simulation.

- **Problem:** Space complexity limits the number of time steps through which gradients can be traced, since space complexity grows linearly with step count.

- **Solution:** Space complexity can be traded for computational complexity through checkpointing, in which a subset of simulation time steps are saved and the intermediate steps are regenerated as needed during the backward pass. For more details see, e.g., Appx. D.2 of [26].

- **Problem:** Maintaining accurate and stable gradients over long periods of simulation time can be challenging due to the accumulation of floating point errors, as well as vanishing and exploding gradients.

- **Solution:** Objective functions could be devised that depend on smaller windows of simulation time and thus do not require backpropagating to the initial time step [63].

B. Limitations of the encoding.

- **Problem:** Encoding the phenotypic details of every body part directly into the genotype scales unfavorably with morphological size and complexity.

- **Solution:** Indirect encodings such as grammars [24], directed graphs [57], pattern-producing networks [9], and others [60] can be used to compress phenotypic complexity into a compact representation.

C. Limitations of the genetic algorithm.

- **Problem:** Neural structures of the robots in this paper were not inheritable. Thus any good behaviors discovered by progenitors during their lifetime (i.e. through gradient based learning) had to be rediscovered in offspring ab initio, which precluded genetic assimilation through Baldwinian effects [33, 23], and which may have limited the robots' behavioral complexity.

- **Solution:** Neural architectures and synaptic weights could be transmitted transgenerationally and co-optimized during the lifetime of a robot. Intergenerational weight sharing could also allow for "culture" to emerge through which "good tricks" are rapidly transmitted across the population.

D. Limitations of the learning algorithm.

- **Problem:** Our naive gradient-based learning procedure is not guaranteed to discover the globally optimal controller for a given morphology. As a result, good body designs may be discarded due to sub-optimal control optimization.

- **Solution:** Although there is no known, provably optimal gradient based-learning scheme, many additional steps can be taken to improve traversal of the learning loss landscape and reduce the likelihood of obtaining substantially sub-optimal performance. These include, but are not limited, to adaptive learning algorithms [29].

E. Limitations of a static environment.

- **Problem:** The conducted evolutionary trials appear to be converging to highly-specialized body plans for the simple and static environment and selection pressures.

- **Solution:** Morphological complexity may be correlated with environmental complexity [2], which could include other agents [50] and competition for resources [57]. Additional niches may be created in the environment by introducing additional search objectives and/or explicitly rewarding diversity [39].

F. Limitations of the variation operators.

- **Problem:** Genetic recombination (crossover) is believed to serve an adaptive function in animals [52]; however, our naive implementation of crossover (as described in Sect. II-H) seems to have slightly inhibited evolutionary innovation as indicated by a "deceleration" of the fitness curves of independent evolutions which employed crossover (Fig. 6) compared to those of evolutions which did not (Fig. 5).

- **Solution:** Methods for recombining neural structures [58] may be adapted for morphological structures under an appropriate genetic encoding (see Sect. IV-B).

G. Limitations of the actuators.

- **Problem:** Two way shape memory alloy springs are not constrained to expand along a linear path. Thus, as the active springs expand they may bend, resulting in a non-linear relationship between spring length and force exerted on its endpoints.

- **Solution:** Alternate actuation mechanisms could be utilized which may allow for more precise control over actuation, and scale more readily to larger numbers of independent actuators in the physical robot. Some examples include tendon driven contractile actuators, pneumatic actuators, lead screw or solenoid based linear actuators.

- **Problem:** Uneven cooling of the spring actuators led to out-of-plane motion response not accounted for in simulation.

- **Solution:** Adding uniform airflow across the entire body, or changing to alternate classes of actuators could further improve strictly planar motion.

H. Limitations of sim2real.

- **Problem:** Physical robots were not able to actuate with the same strength and speed as their simulated counterparts.

- **Solution:** An iterative process of physical system characterization and simulation development could result in a tighter cyber-physical coupling.

V. Conclusion

Thirty years ago, the very first studies to automatically design robots did so by evolving populations of a few hundred morphologically-simple robots, each composed of a dozen or so moving parts. Since then the field has experienced a long period of stasis; despite exponential increases in computational power and the embarrassingly parallel nature of evolutionary algorithms, contemporary work still features comparable population sizes and low sensorimotor complexity (Table I). This lack of progress can largely be attributed to the inefficiencies of the serialized non-differentiable simulations used to evaluate design variants.

In this paper we unified evolutionary robotics with differentiable policy training and, in doing so, enhanced the scale of the former by orders of magnitude. Our results demonstrate a synchronicity between evolutionary morphological search and gradient based controller optimization wherein body plans with increasing fitness potential—by way of learning—are discovered. To demonstrate the power of joining evolution and gradient based learning at scale, we intentionally chose the most simple, "vanilla" algorithmic components wherever possible. Despite these simplifications, we observed the emergence of millions of novel robots with complex anatomical forms (body shapes, musculature, and proprioceptors) that elegantly coordinated their movement across interesting terrains. In this light, these results both highlight the power of joining large-scale evolution with differentiable training and suggest the potential for further advances with this framework.

VI. Code

Supporting code is available at **github.com/lstrgar/ELDiR**.

Acknowledgements

This research was supported by Schmidt Sciences AI2050 grant G-22-64506 and Templeton World Charity Foundation award no. 20650.

References

[1] Joshua E Auerbach and Josh C Bongard. Evolving complete robots with CPPN-NEAT: The utility of recurrent connections. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 1475–1482, 2011.

[2] Joshua E Auerbach and Josh C Bongard. Environmental influence on the evolution of morphological complexity in machines. *PLoS Computational Biology*, 10(1): e1003399, 2014.

[3] Jagdeep Bhatia, Holly Jackson, Yunsheng Tian, Jie Xu, and Wojciech Matusik. Evolution gym: A large-scale benchmark for evolving soft robots. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:2201–2214, 2021.

[4] Josh Bongard and Rolf Pfeifer. Evolving complete agents using artificial ontogeny. In *Morpho-functional machines: The new species: Designing embodied intelligence*, pages 237–258, 2003.

[5] Luzius Brodbeck, Simon Hauser, and Fumiya Iida. Morphological evolution of physical robots through model-free phenotype development. *PloS ONE*, 10(6): e0128444, 2015.

[6] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Tong Wang, Samuel Marks, Charbel-Raphael Segerie, Micah Carroll, Andi Peng,

Phillip Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Biyik, Anca Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open problems and fundamental limitations of reinforcement learning from human feedback. *Transactions on Machine Learning Research*, 2023.

[7] Daniel Cellucci, Robert MacCurdy, Hod Lipson, and Sebastian Risi. 1d printing of recyclable robots. *IEEE Robotics and Automation Letters*, 2(4):1964–1971, 2017.

[8] Nicolas Chaumont, Richard Egli, and Christoph Adami. Evolving virtual creatures and catapults. *Artificial Life*, 13(2):139–157, 2007.

[9] Nick Cheney, Robert MacCurdy, Jeff Clune, and Hod Lipson. Unshackling evolution: Evolving soft robots with multiple materials and a powerful generative encoding. In *Proceedings of the Conference on Genetic and Evolutionary Computation (GECCO)*, pages 167–174. ACM, 2013.

[10] Nick Cheney, Jeff Clune, and Hod Lipson. Evolved electrophysiological soft robots. In *Proceedings of the Conference on Artificial Life (ALife)*, volume 14, pages 222–229, 2014.

[11] Nick Cheney, Josh Bongard, Vytas SunSpiral, and Hod Lipson. On the difficulty of co-optimizing morphology and control in evolved virtual creatures. In *Proceedings of the Artificial life Conference (ALIFE)*, pages 226–233. MIT Press, 2016.

[12] Nick Cheney, Josh Bongard, Vytas SunSpiral, and Hod Lipson. Scalable co-optimization of morphology and control in embodied machines. *Journal of The Royal Society Interface*, 15(143):20170937, 2018.

[13] François Cochevelou, David Bonner, and Martin-Pierre Schmidt. Differentiable soft-robot generation. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, page 129–137. ACM, 2023.

[14] Francesco Corucci, Nick Cheney, Francesco Giorgio-Serchi, Josh Bongard, and Cecilia Laschi. Evolving soft locomotion in aquatic and terrestrial environments: Effects of material properties and environmental transitions. *Soft Robotics*, 5(4):475–495, 2018.

[15] Jonas Degrave, Michiel Hermans, Joni Dambre, et al. A differentiable physics engine for deep learning in robotics. *Frontiers in Neurorobotics*, page 6, 2019.

[16] Adam Gaier and David Ha. Weight agnostic neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

[17] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 249–256. JMLR, 2010.

[18] Agrim Gupta, Silvio Savarese, Surya Ganguli, and Li Fei-Fei. Embodied intelligence via learning and evolution. *Nature Communications*, 12(1):1–12, 2021.

[19] Donald J Hejna III, Pieter Abbeel, and Lerrel Pinto. Task-agnostic morphology evolution. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[20] Michiel Hermans, Benjamin Schrauwen, Peter Bienstman, and Joni Dambre. Automated design of complex dynamic systems. *PloS ONE*, 9(1):e86696, 2014.

[21] Jonathan Hiller and Hod Lipson. Evolving amorphous robots. In *Conference on Artificial Life (ALife)*, pages 717–724, 2010.

[22] Jonathan Hiller and Hod Lipson. Automatic design and manufacture of soft robots. *IEEE Transactions on Robotics*, 28(2):457–466, 2012.

[23] Geoffrey E Hinton and Steven J Nowlan. How learning can guide evolution. *Complex Systems*, 1:495–502, 1987.

[24] Gregory S Hornby, Hod Lipson, and Jordan B Pollack. Generative representations for the automated design of modular physical robots. *IEEE Transactions on Robotics and Automation*, 19(4):703–719, 2003.

[25] Yuanming Hu, Jiancheng Liu, Andrew Spielberg, Joshua B Tenenbaum, William T Freeman, Jiajun Wu, Daniela Rus, and Wojciech Matusik. Chainqueen: A real-time differentiable physical simulator for soft robotics. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 6265–6271. IEEE, 2019.

[26] Yuanming Hu, Luke Anderson, Tzu-Mao Li, Qi Sun, Nathan Carr, Jonathan Ragan-Kelley, and Frédo Durand. Difftaichi: Differentiable programming for physical simulation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[27] Nick Jakobi, Phil Husbands, and Inman Harvey. Noise and the reality gap: The use of simulation in evolutionary robotics. In *Proceedings of the European Conference on Artificial Life (ECAL)*, pages 704–720, 1995.

[28] Michał Joachimczak, Reiji Suzuki, and Takaya Arita. Artificial metamorphosis: Evolutionary design of transforming, soft-bodied robots. *Artificial Life*, 22(3):271–298, 2016.

[29] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[30] Maciej Komosiński and Adam Rotaru-Varga. Comparison of different genotype encodings for simulated three-dimensional agents. *Artificial Life*, 7(4):395–418, 2001.

[31] John R. Koza. *Genetic programming: on the programming of computers by means of natural selection*. MIT Press, Cambridge, MA, USA, 1992.

[32] Peter Krčah. Solving deceptive tasks in robot body-brain co-evolution by searching for behavioral novelty. In *Proceedings of the International Conference on Intelligent Systems Design and Applications*, pages 284–289. IEEE, 2010.

[33] Sam Kriegman, Nick Cheney, and Josh Bongard. How morphological development can guide evolution. *Scien-*

tific Reports, 8(1):13934, 2018.

[34] Sam Kriegman, Nick Cheney, Francesco Corucci, and Josh C. Bongard. Interoceptive robustness through environment-mediated morphological development. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 109–116. ACM, 2018. URL https://arxiv.org/abs/1804.02257.

[35] Sam Kriegman, Douglas Blackiston, Michael Levin, and Josh Bongard. A scalable pipeline for designing reconfigurable organisms. *Proceedings of the National Academy of Sciences*, 117(4):1853–1859, 2020.

[36] Sam Kriegman, Amir Mohammadi Nasab, Dylan Shah, Hannah Steele, Gabrielle Branin, Michael Levin, Josh Bongard, and Rebecca Kramer-Bottiglio. Scalable sim-to-real transfer of soft robot designs. In *Proceedings of the International Conference on Soft Robotics (RoboSoft)*, pages 359–366, 2020.

[37] Sam Kriegman, Douglas Blackiston, Michael Levin, and Josh Bongard. Kinematic self-replication in reconfigurable organisms. *Proceedings of the National Academy of Sciences*, 118(49):e2112672118, 2021.

[38] Sam Kriegman, Amir Mohammadi Nasab, Douglas Blackiston, Hannah Steele, Michael Levin, Rebecca Kramer-Bottiglio, and Josh Bongard. Scale invariant robot behavior with fractals. In *Robotics: Science and Systems (RSS)*, 2021.

[39] Joel Lehman and Kenneth O Stanley. Evolving a diversity of virtual creatures through novelty search and local competition. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 211–218, 2011.

[40] Richard E Lenski, Charles Ofria, Robert T Pennock, and Christoph Adami. The evolutionary origin of complex features. *Nature*, 423(6936):139–144, 2003.

[41] Dan Lessin, Don Fussell, and Risto Miikkulainen. Open-ended behavioral complexity for evolved virtual creatures. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 335–342, 2013.

[42] Muhan Li, David Matthews, and Sam Kriegman. Reinforcement learning for freeform robot design. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2024.

[43] Hod Lipson and Jordan B Pollack. Automatic design and manufacture of robotic lifeforms. *Nature*, 406(6799):974, 2000.

[44] Pingchuan Ma, Tao Du, John Z Zhang, Kui Wu, Andrew Spielberg, Robert K Katzschmann, and Wojciech Matusik. Diffaqua: A differentiable computational design pipeline for soft underwater swimmers with shape interpolation. *ACM Transactions on Graphics (TOG)*, 40 (4):1–14, 2021.

[45] David Matthews, Andrew Spielberg, Daniela Rus, Sam Kriegman, and Josh Bongard. Efficient automatic design of robots. *Proceedings of the National Academy of Sciences*, 120(41):e2305180120, 2023.

[46] Eric Medvet, Alberto Bartoli, Federico Pigozzi, and Marco Rochelli. Biodiversity in evolved voxel-based soft robots. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 129–137, 2021.

[47] Thomas Miconi and Alastair Channon. An improved system for artificial creatures evolution. In *Proceedings of the International Conference on the Simulation and Synthesis of Living Systems (ALife)*, 2006.

[48] Karine Miras, Eliseo Ferrante, and AE Eiben. Environmental influences on evolvable robots. *PloS ONE*, 15(5): e0233848, 2020.

[49] Rodrigo Moreno and Andres Faiña. EMERGE modular robot: a tool for fast deployment of evolved robots. *Frontiers in Robotics and AI*, 8:699814, 2021.

[50] Stefano Nolfi and Dario Floreano. Coevolving predator and prey robots: Do "arms races"' arise in artificial evolution? *Artificial Life*, 4(4):311–335, 1998.

[51] Emma Stensby Norstein, Frank Veenstra, Kai Olav Ellefsen, Tønnes Nygaard, and Kyrre Glette. Effects of compliant and structural parts in evolved modular robots. In *Proceedings of the Artificial Life Conference (ALife)*. MIT Press, 2023.

[52] Sarah P Otto. The evolutionary enigma of sex. *The American Naturalist*, 174(S1):S1–S14, 2009.

[53] Deepak Pathak, Christopher Lu, Trevor Darrell, Phillip Isola, and Alexei A Efros. Learning to control self-assembling morphologies: a study of generalization via modularity. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[54] Federico Pigozzi, Eric Medvet, Alberto Bartoli, and Marco Rochelli. Factors impacting diversity and effectiveness of evolved modular robots. *ACM Transactions on Evolutionary Learning (TELO)*, 3(1):1–33, 2023.

[55] John Rieffel, Davis Knox, Schuyler Smith, and Barry Trimmer. Growing and evolving soft robots. *Artificial Life*, 20(1):143–162, 2014.

[56] Charles Schaff, Audrey Sedal, and Matthew R Walter. Soft robots learn to crawl: Jointly optimizing design and control with sim-to-real transfer. In *Robotics: Science and Systems (RSS)*, 2022.

[57] Karl Sims. Evolving 3D morphology and behavior by competition. *Artificial Life*, 1(4):353–372, 1994.

[58] Kenneth O Stanley and Risto Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2):99–127, 2002.

[59] Merel van Diepen and Kristina Shea. Co-design of the morphology and actuation of soft robots for locomotion. *Journal of Mechanical Design*, 144(8):083305, 2022.

[60] Frank Veenstra and Kyrre Glette. How different encodings affect performance and diversification when evolving the morphology and control of 2D virtual creatures. In *Proceedings of the Conference on Artificial Life (ALife)*, pages 592–601, 2020.

[61] Jeffrey Ventrella. Explorations in the emergence of morphology and locomotion behavior in animated characters.

In *Proceedings of the International Workshop on the Synthesis and Simulation of Living Systems (ALife)*, pages 436–441, 1994.

[62] Tingwu Wang, Yuhao Zhou, Sanja Fidler, and Jimmy Ba. Neural graph evolution: Towards efficient automatic robot design. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

[63] Jie Xu, Viktor Makoviychuk, Yashraj Narang, Fabio Ramos, Wojciech Matusik, Animesh Garg, and Miles Macklin. Accelerated policy learning with parallel differentiable simulation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[64] Ye Yuan, Yuda Song, Zhengyi Luo, Wen Sun, and Kris Kitani. Transform2Act: Learning a transform-and-control policy for efficient agent design. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

[65] Changyoung Yuhn, Yuki Sato, Hiroki Kobayashi, Atsushi Kawamoto, and Tsuyoshi Nomura. 4D topology optimization: Integrated optimization of the structure and self-actuation of soft bodies for dynamic motions. *Computer Methods in Applied Mechanics and Engineering*, 414:116187, 2023.

[66] Allan Zhao, Jie Xu, Mina Konaković-Luković, Josephine Hughes, Andrew Spielberg, Daniela Rus, and Wojciech Matusik. Robogrammar: graph grammar for terrain-optimized robot design. *ACM Transactions on Graphics (TOG)*, 39(6):1–16, 2020.