Unshackling Evolution: Evolving Soft Robots with Multiple Materials and a Powerful Generative Encoding

Nick Cheney, Robert MacCurdy, Jeff Clune*, Hod Lipson
Creative Machines Lab, Cornell University
Ithaca, New York, USA
*Evolving AI Lab, University of Wyoming
*Laramie, Wyoming, USA
nac93@cornell.edu, rbm7@cornell.edu, jeffclune@uwyo.edu, hod.lipson@cornell.edu

ABSTRACT

In 1994 Karl Sims showed that computational evolution can produce interesting morphologies that resemble natural organisms. Despite nearly two decades of work since, evolved morphologies are not obviously more complex or natural, and the field seems to have hit a complexity ceiling. One hypothesis for the lack of increased complexity is that most work, including Sims', evolves morphologies composed of rigid elements, such as solid cubes and cylinders, limiting the design space. A second hypothesis is that the encodings of previous work have been overly regular, not allowing complex regularities with variation. Here we test both hypotheses by evolving *soft robots* with *multiple materials* and a *powerful generative encoding* called a compositional pattern-producing network (CPPN). Robots are selected for locomotion speed. We find that CPPNs evolve faster robots than a direct encoding and that the CPPN morphologies appear more natural. We also find that locomotion performance increases as more materials are added, that diversity of form and behavior can be increased with different cost functions without stifling performance, and that organisms can be evolved at different levels of resolution. These findings suggest the ability of generative soft-voxel systems to scale towards evolving a large diversity of complex, natural, multi-material creatures. Our results suggest that future work that combines the evolution of CPPN-encoded soft, multi-material robots with modern diversity-encouraging techniques could finally enable the creation of creatures far more complex and interesting than those produced by Sims nearly twenty years ago.

Categories and Subject Descriptors: I.2.11 [**Distributed Artificial Intelligence**]:Intelligent Agents

General Terms: Algorithms, Design, Experimentation
Keywords: Genetic Algorithms, Generative Encodings, CPPN-NEAT, Soft-Robotics, HyperNEAT, Evolving Morphologies

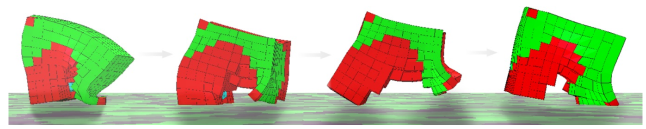

Figure 1: An example of a natural looking morphology and behavior evolved by combining a generative encoding with voxel-resolution soft, actuatable materials. The soft robot gallops from left to right across the image with a dog-like gait.

1. INTRODUCTION

In 1994, Karl Sims' evolved virtual creatures showed the potential of evolutionary algorithms to produce natural, complex morphologies and behaviors [30]. One might assume that nearly 20 years of improvements in computational speed and evolutionary algorithms would produce far more impressive organisms, yet the creatures evolved in the field of artificial life today are not obviously more complex, natural, or intelligent. Fig. 2 demonstrates an example of similar complexity in robots evolved 17 years apart.

One hypothesis for why there has not been a clear increase in evolved complexity is that most studies follow Sims in evolving morphologies with a limited set of rigid elements [21, 4, 3, 16, 22]. Nature, in contrast, composes organisms with a vast array of different materials, from soft tissue to hard bone, and uses these materials to create subcomponents of arbitrary shapes. The ability to construct morphologies with heterogeneous materials enables nature to produce more complex, agile, high-performing bodies [35]. An open question is whether computational evolution will

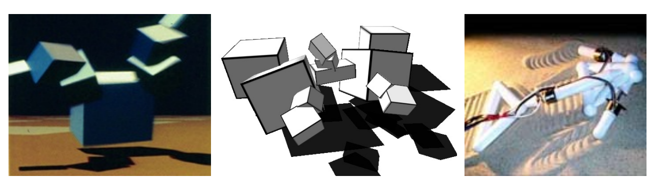

Figure 2: (left) The scale and resolution of robots evolved by Sims in 1994 [30]. (middle) The scale and resolution at which evolutionary robotics commonly occurs today (from Lehman and Stanley, 2011 [21]). (right) The scale and resolution of robot fabrication techniques (from Lipson and Pollack, 2000 [22]).

produce more natural, complex forms if it is able to create organisms out of many material types. Here we test that hypothesis by evolving morphologies composed of voxels of different materials. They can be hard or soft, analogous to bone or soft tissue, and inert or expandable, analogous to supportive tissue or muscle. Contiguous patches of homogeneous voxels can be thought of as different tissue structures.

Another hypothesis is that the encodings used in previous work limited the design space. Direct encodings lack the regularity and evolvability necessary to consistently produce regular morphologies and coordinated behaviors [9, 6, 34, 16], and overly regular indirect encodings constrict the design space by disallowing complex regularities with variation [16, 31, 34]. We test this hypothesis by evolving morphologies with the CPPN-NEAT encoding [31], which has been shown to create complex regularities such as symmetry and repetition, both with and without variation (Fig. 3). CPPN-NEAT has shown these abilities in 2D images [29] and 3D objects [7] and morphologies [4]. To test the impact of the CPPN encoding, we compare it to a direct encoding.

Overall, we find that evolution does utilize additional materials made available to it; their availability led to a significant amount of diverse, interesting, complex morphologies and locomotion behaviors without hindering performance. Furthermore, the generative encoding produced regular patterns of voxel 'tissue', leading to fast, effective locomotion. In contrast, the direct encoding produced no phenotypic regularity and led to poor performance.

Because it is notoriously difficult to quantify attributes such as "impressiveness" and "complexity", we make no effort to do so here. Instead, we attempt to visually represent the interesting diversity of morphologies and behaviors that evolved once evolution was provided with more materials and a sophisticated encoding. We also demonstrate the ability for this system to scale to higher resolutions and greater material diversity without hindering performance. Finally, we investigate the effects of different fitness functions, revealing that evolution with this encoding and material palette can create different bodies and behaviors in response to different environmental and selective pressures.

2. BACKGROUND

There are many Evolutionary Robotics papers with rigid-body robots [25]. However, few attempts have been made to evolve robots composed of soft materials [27], and most of those attempts are limited to only a few components. This paucity is due largely to the computational costs of simulating flexible materials and because many genetic encodings do not scale to large parameter spaces [5, 18].

The CPPN encoding abstracts how developmental biology builds natural complexity, and has been shown to produce complex, natural-appearing images and objects (Fig. 3) [29, 7, 31]. Auerbach and Bongard used this generative encoding to evolve robotic structures at finer resolutions than previous work. The systems evolved demonstrated the ability to take advantage of geometric coordinates to inform the evolution of complex bodies. However, this work was limited to rigid building blocks which were actuated by a large number of hinge joints [1, 4, 3], or had no actuation at all [2].

Rigid structures limit the ability of robots to interact with their environments, especially when compared to the complex movements of structures in biology composed of muscle and connective tissue. These structures, called muscular

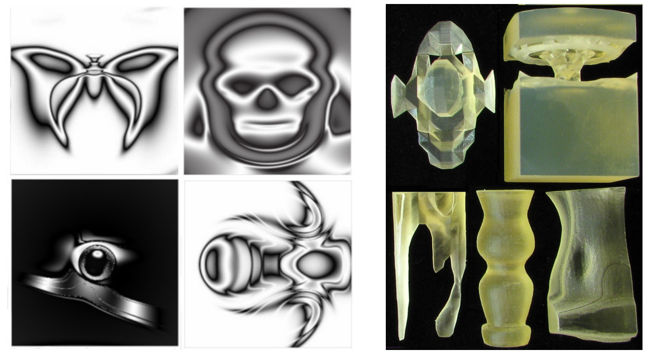

Figure 3: (left) Examples of high resolution, complex, natural-looking images evolved with CPPN-NEAT that contain symmetry, repetition, and interesting variation [29]. (right) Examples of CPPN-encoded 3D shapes with these same properties [7]).

hydrostats, often display incredible flexibility and strength; examples from biology include octopus arms or elephant trunks [35]. While soft robots can be designed that provide outstanding mobility, strength and reliability, the design process is complicated by multiple competing and difficult-to-define objectives [35]. Evolutionary algorithms excel at such problems, but have historically not been able to scale to larger robotic designs. To demonstrate that evolution can design complex, soft-bodied robots, Hiller and Lipson created a soft-voxel simulator (called VoxCAD) [11]. They showed a preliminary result that CPPNs can produce interesting locomotion morphologies, and that such designs can transfer to the real world (Fig. 4) [13]. However, this work did not take advantage of the NEAT algorithm, with its historical markings, speciation, crossover, and complexification over time - which have been shown to greatly improve the search process [33]. Additionally, these preliminary results consisted of only three trials per treatment. Here we conduct a more in-depth exploration of the capabilities of CPPNs when evolving soft robots in VoxCad.

Figure 4: A time-series example of a fabricated soft robot, which actuates with cyclic 20% volumetric actuation in a pressure chamber [13]. This proof-of-concept shows that evolved, soft-bodied robots can be physically realized. Current work is investigating soft robot actuation outside of a pressure chamber.

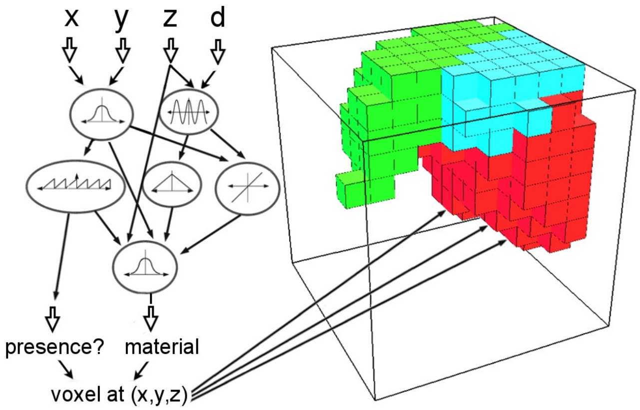

Figure 5: A CPPN is iteratively queried for each voxel within a bounding area and produces output values as a function of the coordinates of that voxel. These outputs determine the presence of voxels and their material properties to specify a soft robot.

3. METHODS

3.1 CPPN-NEAT

CPPN-NEAT has been repeatedly described in detail [31, 9, 7, 10], so we only briefly summarize it here. A compositional pattern-producing network (CPPN) is similar to a neural network, but its nodes contain multiple math functions (in this paper: sine, sigmoid, Gaussian, and linear). CPPNs evolve according to the NEAT algorithm [31]. The CPPN produces geometric output patterns that are built up from the functions of these nodes. Because the nodes have regular mathematical functions, the output patterns tend to be regular (e.g. a Gaussian function can create symmetry and a sine function can create repetition). In this paper, each voxel has an x, y, and z coordinate that is input into the network, along with the voxel's distance from center (d). One output of the network specifies whether any material is present, while the maximum value of the 4 remaining output nodes (each representing an individual material) specifies the type of material present at that location (Fig. 5). This method of separating the presence of a phenotypic component and its parameters into separate CPPN outputs has been shown to improve performance [36]. Robots can be produced at any desired resolution. If there are multiple disconnected patches, only the most central patch is considered when producing the robot morphology.

3.2 VoxCAD

Fitness evaluations are performed in the VoxCAD soft-body simulator, which is described in detail in Hiller and Lipson 2012 [14]. The simulator efficiently models the statics, dynamics, and non-linear deformation of heterogeneous soft bodies. It also provides support for volumetric actuation of individual voxels (analogous to expanding and contracting muscles) or passive materials of varying stiffness (much like soft support tissue or rigid bone). For visualization, we display each voxel, although a smooth surface mesh could be added via the Marching Cubes algorithm [23, 7].

3.2.1 MATERIALS

Following [12], there are two types of voxels: those that expand and contract at a pre-specified frequency, and passive voxels with no intrinsic actuation, which are either soft or hard. We expand upon [12] to include multiple phases of actuation. Unless otherwise noted, four materials are used: Green voxels undergo periodic volumetric actuations of 20%. Light blue voxels are soft and passive, having no intrinsic actuation, with their deformation caused solely by nearby voxels. Red voxels behave similarly to green ones, but with counter-phase actuations. Dark blue voxels are also passive, but are more stiff and resistant to deformation than light blue voxels. In treatments with less than 4 materials, voxels are added in the order above (e.g. two material treatments consist of green and light blue voxels).

3.3 GAlib

The direct encoding is from GAlib–fully described in [37]–a popular off-the-shelf genetic algorithm library from MIT. In the direct encoding genome, each voxel has its own independent values representing its presence and material outputs. The first value is binary, indicating whether a voxel at that position exists. If the voxel exists, the highest of the material property values determines the type of voxel. Thus, a $10 \times 10 \times 10$ ("10^3") voxel soft robot with 4 possible materials would have a genome size of $10^3 \times 5 = 5000$ values.

3.4 Experimental Details

Treatments consist of 35 runs, each with a population size of 30, evolved for 1000 generations. Unless otherwise noted, fitness is the difference in the center of mass of the soft robot between initialization and the end of 10 actuation cycles. If any fitness penalties are assessed, they consist of multiplying the above fitness metric by: $1 - \frac{\text{penalty metric}}{\text{maximum penalty metric}}$. For example, if the penalty metric is the number of voxels, an organism with 400 non-empty voxels out of a possible 1000 would have its displacement multiplied by $1 - \frac{400}{1000} = 0.6$ to produce its final fitness value. Other CPPN-NEAT parameters are the same as in Clune and Lipson 2011 [7].

4. RESULTS

Quantitative and qualitative analyses reveal that evolution in this system is able to produce effective and interesting locomoting soft robots at different voxel resolutions and using different materials. We also discover that imposing different environmental challenges in the form of penalty functions provides an increased diversity of forms, suggesting the capability to adapt to various selective pressures.

Videos of soft robot locomotion are available at `http://tinyurl.com/EvolvingSoftRobots`. So the reader may verify our subjective, qualitative assessments, we have permanently archived all evolved organisms, data, source code, and parameter settings at the Dryad Digital Repository.

4.1 Direct vs. Generative Encoding

The CPPN-NEAT generative encoding far outperforms the direct encoding (Figure 8), which is consistent with previous findings [9, 6]. The most stark difference is in the regularity of the voxel distributions (compare Figs. 1, 6, 12, 13 to Fig. 7). CPPN-NEAT soft robots consist of homogeneous patches of materials akin to tissues (e.g. one large patch of muscle, another patch of bone, etc.). The direct encoding,

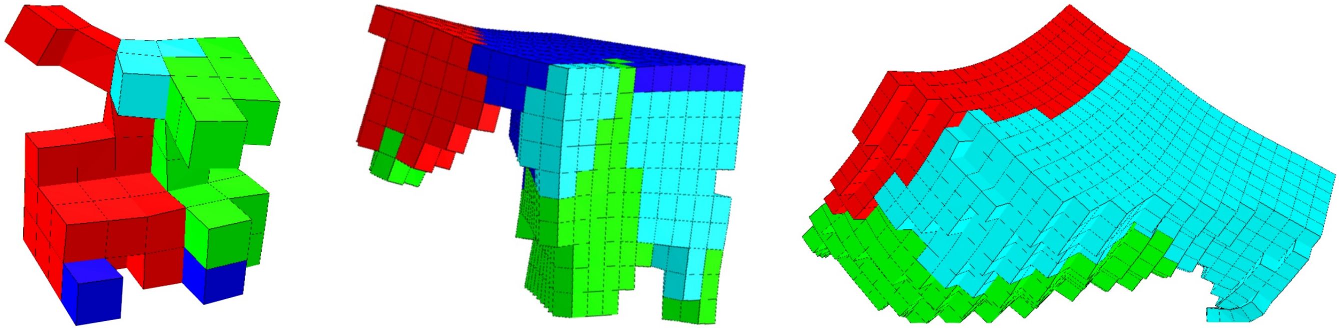

Figure 6: CPPN-NEAT-encoded soft robots can scale to any resolution. Pictured here are soft robots sampled at voxel resolutions of $5 \times 5 \times 5$ (left), $10 \times 10 \times 10$ (center), and $20 \times 20 \times 20$ (right).

on the other hand, seems to randomly assign a material to each voxel. These homogeneous tissue structures are beneficial because similar types of voxels can work in a coordinated fashion to achieve the locomotion objective. For example, all the voxels in one large section of green voxels will expand at the same time, functioning as muscle tissue. This global coordination leads to jumping, bounding, stepping, and many other behaviors. In the direct encoding, each voxel works independently from–and often at odds with–its neighboring voxels, preventing coordinated behaviors. Instead, final organisms appear visually similar to those at initialization, and performance barely improves across generations (Figure 8).

Another reason for the success of the CPPN-NEAT encoding is one of the key properties of the NEAT algorithm: it starts with CPPN networks that produce simple geometric voxel patterns and *complexifies* those patterns over time [31].

4.2 Penalty Functions

To explore performance under different selective or environmental pressures, we tested four different penalty regimes. All four require the soft robot to move as far as possible, but have different restrictions. In one environment, the soft robots are penalized for their number of voxels, similar to an animal having to work harder to carry more weight. In another, the soft robots are penalized for their amount of actuatable material, analogous to the cost of expending energy to contract muscles. In a third treatment, a penalty is assessed for the number of connections (adjoining faces between voxels), akin to animals that live in warm environments and overheat if their surface area is small in comparison to their volume. Finally, there is also the baseline treatment in which no penalties are assessed.

While a cost for actuated voxels does perform significantly worse than a setup with no cost ($p = 1.9 \times 10^{-5}$ comparing final fitness values), all treatments tend to perform similarly over evolutionary time (Fig. 9). This rough equivalence suggests that the system has the ability to adapt to different cost requirements without major reductions in performance. However, drastically different types of body-plans and behaviors evolved for the different fitness functions. There are differences in the proportions of each material found in evolved organisms, indicating that evolution utilizes different material distributions to fine tune morphologies to various environments (Fig. 10). For example, when no penalty cost is assessed, more voxels are present ($p < 2 \times 10^{-13}$). When there is a cost for the number of actuated voxels, but not for support tissue, evolution uses more of these inert support materials ($p < 0.02$).

More revealing are the differences in behaviors. Fig. 11 categorizes locomotion strategies into several broad classes, and shows that different task requirements favor different classes of these behaviors. To limit subjectivity in the categorization process, we made clear category definitions, as is common in observational biology, and provide an online archive of all organisms for reader evaluation (see Sec. 4).

Fig. 12 displays the common locomotion strategies and Fig. 11 shows how frequently they evolved. They are described in order of appearance in Fig. 12. The L-Walker is named after the “L” shape its rectangular body forms, and is distinguished by its blocky form and hinge-like pivot point in the bend of the L. The Incher is named after its inchworm like behavior, in which it pulls its back leg up to its front legs by arching its back, then stretches out to flatten itself and reach its front legs forward. Its morphology is distinguished by its sharp spine and diagonal separation between actuatable materials. The Push-Pull is a fairly wide class of behaviors and is tied together by the soft robot’s powerful push with its (often large) hind leg to propel itself forward, which is usually coupled with a twisting or tipping of its front limb/head to pull itself forward between pushes. The head shape and thinner neck region are surprisingly common features. Next, the Jitter (or Bouncer) moves by bouncing

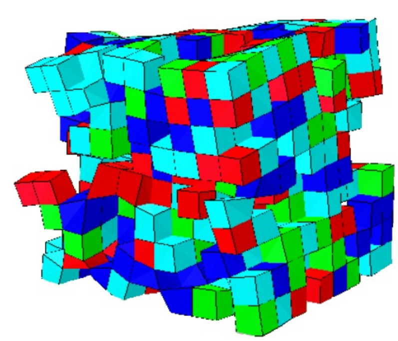

Figure 7: A representative example of a soft robot evolved with a direct encoding. Note the lack of regularity and organization: there are few contiguous, homogeneous patches of one type of voxel. Instead, the organism appears to be composed of randomly distributed voxels . The resolution is the default 10^3.

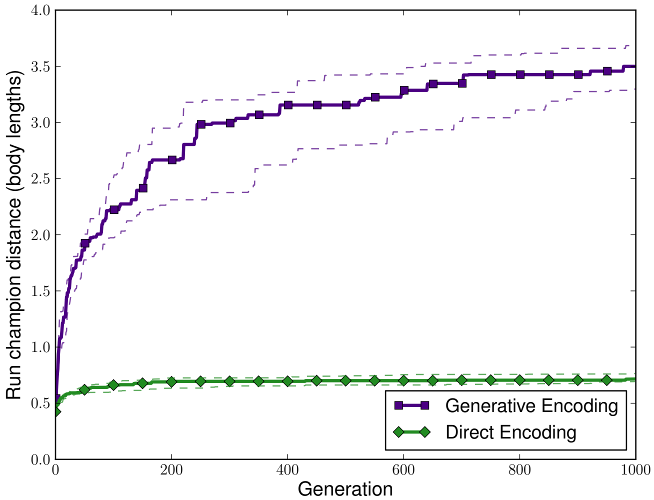

Figure 8: The best individuals from 35 independent runs with a direct or generative encoding. Note how the generative encoding sees large improvements early in evolution, while it is exploring new locomotion types. It then settles on specific types and gradually improves coordination, timing, etc., to exploit a given strategy. The direct encoding is unable to produce globally coordinated behavior to develop new locomotion strategies, resulting in very minor improvements as it exploits its initial random forms. Here, and in all figures, thick lines are medians ±95% bootstrapped confidence intervals.

its (often large) back section up and down, which pushes the creature forward. It is distinguished by its long body and is often composed mainly of a single actuatable material. The Jumper is similar in that it is often comprised of a single actuatable material, but locomotes in an upright position, springing up into the air and using its weight to angle its jumping and falling in a controlled fashion to move forward. The Wings is distinguished by its unique vertical axis of rotation. It brings its arms (or wings) in front of it, then pushes them down and out to the sides, propelling its body forward with each flapping-like motion. Fig. 13 demonstrates other, less-common behaviors that evolved.

These example locomotion strategies display the system's ability to produce a diverse set of morphologies and behaviors, which likely stems from its access to multiple types of materials. Our results suggest that with even more materials, computational evolution could produce even more sophisticated morphologies and behaviors. Note that different behaviors show up more frequently for different task settings (Fig. 11), suggesting the ability of the system to fine tune to adapt to different selective pressures.

4.3 Material Types

To meet its full potential, this system must scale to arbitrarily large numbers of materials and resolutions. We first explore its ability to compose soft robots out of a range of materials by separately evolving soft robots with increasing numbers of materials (in the order outlined in Sec. 3.2.1).

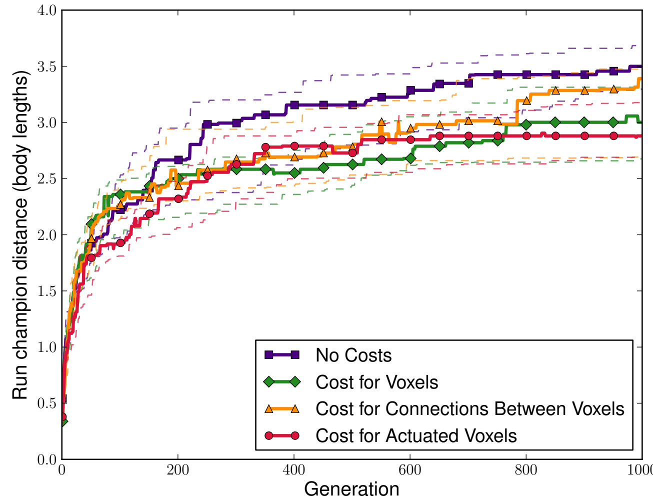

Figure 9: Performance is mostly unaffected by different selection pressures (i.e. fitness functions).

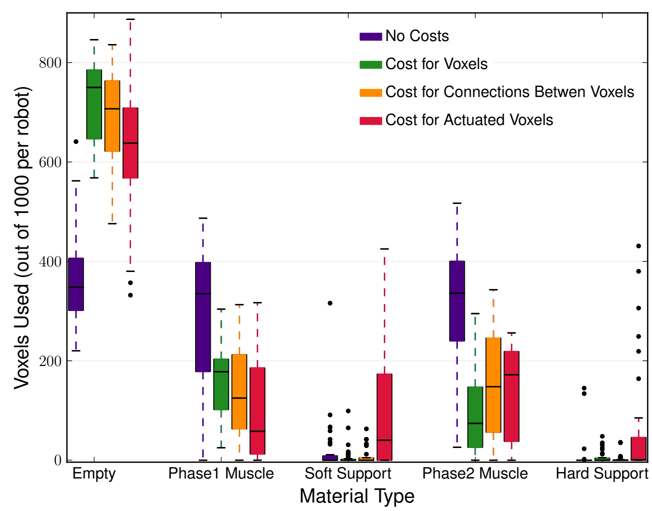

Figure 10: The amount of each material that evolved for different cost functions, revealing the system's ability to adapt material distributions to different environments. For example, without a cost, evolution used more voxels to produce actuation ($p < 2 \times 10^{-13}$). With a cost for actuated voxels, evolution tends to use more inert support tissue ($p < 0.02$).

Adding a second, and then a third, material significantly improved performance (Fig. 14, $p < 2 \times 10^{-6}$), and adding a further hard, inert material did not significantly hurt performance (Fig. 14, $p = 0.68$). This improved performance suggests that CPPN-NEAT is capable of taking advantage of the increase in morphological and behavioral options. This result is interesting, as one might have expected a drop in performance associated with the need to search in a higher dimensional space and coordinate more materials.

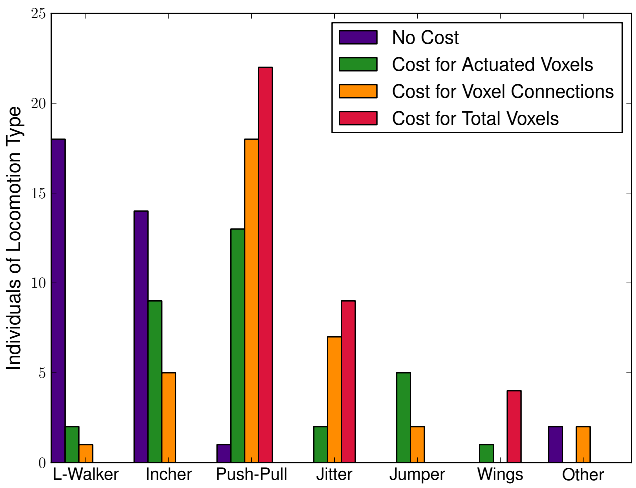

Figure 11: Common behaviors evolved under different cost functions, summed across all runs. These behaviors are described in Sec. 4.2 and visualized in Fig. 12. Some behaviors occur more frequently under certain selective regimes. For example, the L-Walker is more common without a voxel cost, while Jitter, Jumper, and Wings do not evolve in any of the no cost runs.

4.4 Resolution

This system also is capable of scaling to higher resolution renderings of soft robots, involving increasing numbers of voxels. Fig. 6 shows example morphologies evolved at each resolution. The generative encoding tended to perform roughly the same regardless of resolution, although the computational expense of simulating large numbers of voxels prevented a rigorous investigation of the effect of resolution on performance. Faster computers will enable such research and the evolution of higher-resolution soft robots.

5. DISCUSSION

The results show that life-like, complex, interesting morphologies and behaviors are possible when we expand the design space of evolutionary robotics to include soft materials that behave similarly to organic tissue or muscle, and search that design space with a powerful generative encoding like CPPN-NEAT. Our preliminary experiments suggest that soft robotics at the voxel resolution will someday provide complex and breathtaking demonstrations of lifelike artificial forms. Soft robotics will also showcase the ability of evolutionary design because human intuitions and engineering fare poorly in such entangled, non-linear design spaces.

We challenged multiple scientists to design fast, locomoting soft robots by hand, using the same resolution and materials. While the sample size is not sufficient to report hard data, all participants (both those with and without engineering backgrounds) were unable to produce organisms that scored higher than the evolved creatures. Participants noted the surprising difficulty of producing efficient walkers with these four materials. This preliminary experiment supports the claim that systems like the CPPN-NEAT generative encoding will increasingly highlight the effectiveness of automated design relative to a human designer.

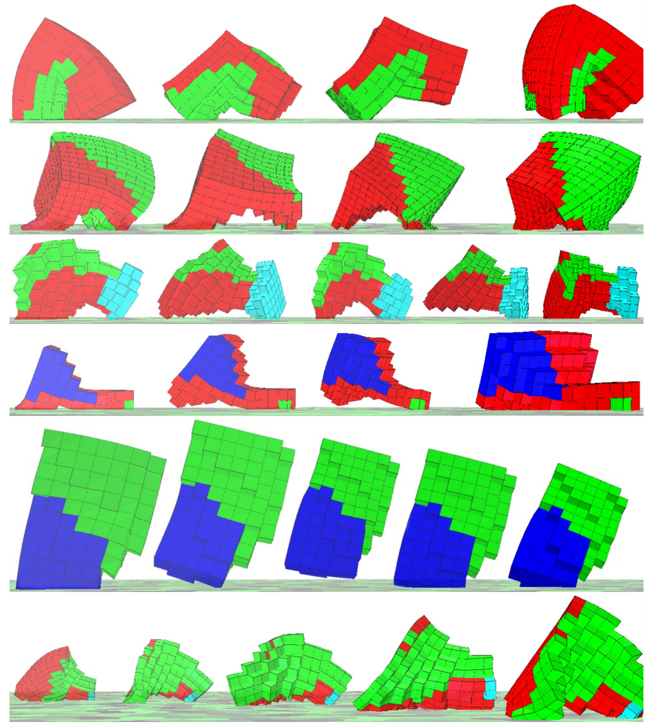

Figure 12: Time series of common soft robot behaviors as they move from left to right across the image. From top to bottom, we refer to them as L-Walker, Incher, Push-Pull, Jitter, Jumper, and Wings. Fig. 11 reports how frequently they evolved.

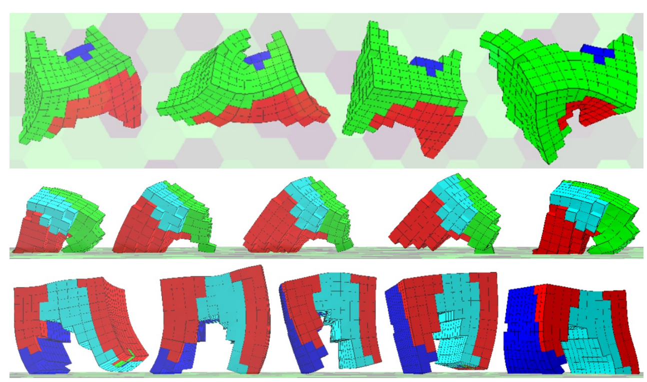

Figure 13: Time series of other evolved strategies. (top) Opposite leg stepping creates a traditional animal walk or trot. (middle) A trunk-like appendage on the front of the robot helps to pull it forward. (bottom) A trot, quite reminiscent of a galloping horse, demonstrates the inclusion of stiff material to create bone-like support in longer appendages.

This work shows that the presence of soft materials alone is not sufficient to provide interesting and efficient locomotion, as soft robots created from the direct encoding performed poorly. Our results are consistent with work evolving rigid-body robots that shows that generative encodings out-

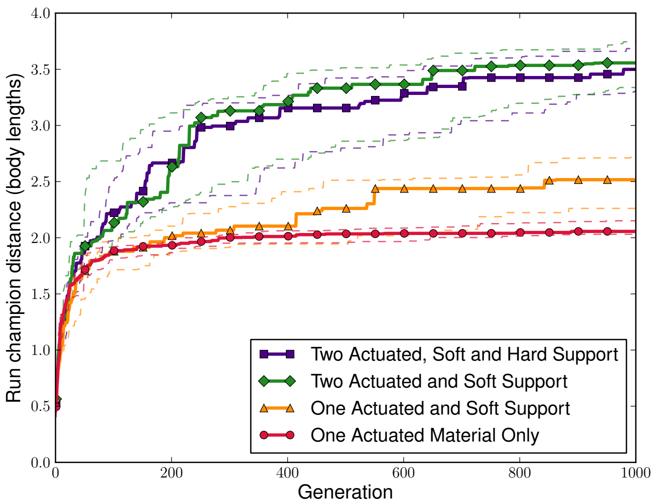

Figure 14: The number of materials also affects performance. With only one, only simple behaviors like Jumping or Bouncing are possible, so performance peaks early and fails to discover new gaits over time. Upon adding a second material, more complex jumping and L-Walker behavior develops. When a second actuatable material is added, most behavior strategies from Fig. 12 become possible. Adding a stiff support material broadens the range of possible gaits, but is only rarely taken advantage of (such as in the bottom gallop of Fig. 13) and thus has a minimal impact on overall performance. These observational assessments may be verified, as all evolved organisms are available online (Sec. 4)

perform direct encodings for evolutionary robotics [17, 19, 9, 6]. Unfortunately, there have been few attempts to evolve robot morphologies with CPPN-NEAT [2], and there is no consensus in the field of a proper measurement of "complexity", "interestingness", or "natural" appearance, so we cannot directly compare our soft robots to their rigid-body counterparts. However, we hope that the reader will agree about the potential of evolved soft robots upon viewing the creatures in action [`http://tinyurl.com/EvolvingSoftRobots`].

6. FUTURE WORK

The ability to evolve complex and intricate forms lends itself naturally to other questions in the field. Auerbach and Bongard have explored the relationship between environment and morphology with rigid robots in highly regular environments [4]. Because our system allows more flexibility in robot morphology and behavior, it may shed additional, or different, light on the relationship between morphology, behavior, and the environment. Preliminary results demonstrate the ability of this system to produce morphologies well suited for obstacles in their environments (Fig. 15).

While our research produced an impressive array of diverse forms, it did use a target-based fitness objective, which can hinder search [38]. Switching to modern techniques for explicitly generating diversity, such as the MOLE algorithm by Mouret and Clune [24, 8] or algorithms by Lehman and

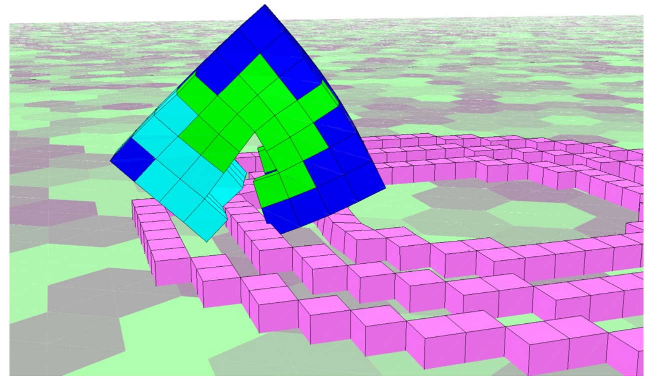

Figure 15: An example of a soft robot that has evolved "teeth" to hook onto the obstacle rings in its environment and propel itself across them.

Stanley [21], has the potential to create an incredibly complex and diverse set of morphologies and behaviors.

Additionally, we are currently pursuing methods to minimize the need for expensive simulations and to evolve specific material properties instead of having a predefined palette of materials. These avenues are expected to allow increased complexity and diversity in future studies.

The HyperNEAT algorithm [32], which utilizes CPPNs, has been shown to be effective for evolving artificial neural network controllers for robots [9, 20, 6]. The same encoding from this work could thus co-evolve robot controllers and soft robot morphologies. Bongard and Pfeifer have argued that such body-brain co-evolution is critical toward progress in evolutionary robotics and artificial intelligence [26].

Soft robots have shown promise in multiple areas of robotics, such as gripping [15] or human-robot interaction [28]. The scale-invariant encoding and soft actuation from this work has potential in these other areas of soft robotics as well.

In order to compare different approaches, the field would benefit from general, accepted definitions and quantitative measures of complexity, impressiveness, and naturalness. Such metrics will enable more quantitative analyses in future work.

7. CONCLUSION

In this work we investigate the difficult-to-address question of why we as a field have failed to substantially improve upon the work of Karl Sims nearly two decades ago. We show that combining a powerful generative encoding based on principles of developmental biology with soft, biologically-inspired materials produces a diverse array of interesting morphologies and behaviors. The evolved organisms are qualitatively different from those evolved in previous research with more traditional rigid materials and either direct, or overly regular, encodings. The CPPN-NEAT encoding produces complex, life-like organisms with properties seen in natural organisms, such as symmetry and repetition, with and without variation. Further, it adapts to increased resolutions, numbers of available materials, and different environmental pressures by tailoring designs to different selective pressures without substantial performance degradation. Our results suggest that investigating soft robotics and modern generative encodings may offer a path towards eventually producing the next generation of impressive, computationally evolved creatures to fill artificial worlds and showcase the power of evolutionary algorithms.

8. ACKNOWLEDGMENTS

NSF Postdoctoral Research Fellowship in Biology to Jeff Clune (DBI-1003220), NSF Graduate Research Fellowship (DGE-0707428) to Robert MacCurdy, and DARPA Grant W911NF-12-1-0449. Thanks also to Jon Hiller for help with the VoxCad simulator.

9. REFERENCES

[1] J. E. Auerbach and J. C. Bongard. Dynamic resolution in the co-evolution of morphology and control. In *Artificial Life XII: Proc. of the 12th International Conf. on the Simulation and Synthesis of Living Systems*, 2010.

[2] J. E. Auerbach and J. C. Bongard. Evolving CPPNs to grow three-dimensional physical structures. In *Proc. of the Genetic & Evolutionary Computation Conf.*, pages 627–634. ACM, 2010.

[3] J. E. Auerbach and J. C. Bongard. On the relationship between environmental and mechanical complexity in evolved robots. In *Proc. of the Artificial Life Conf.*, pages 309–316, 2012.

[4] J. E. Auerbach and J. C. Bongard. On the relationship between environmental and morphological complexity in evolved robots. In *Proc. of the Genetic & Evolutionary Computation Conf.*, pages 521–528. ACM, 2012.

[5] J. Bongard. Evolving modular genetic regulatory networks. In *Evolutionary Computation, 2002. Proc. of the 2002 Congress on*, volume 2, pages 1872–1877. IEEE, 2002.

[6] J. Clune, B. Beckmann, C. Ofria, and R. Pennock. Evolving coordinated quadruped gaits with the HyperNEAT generative encoding. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 2764–2771, 2009.

[7] J. Clune and H. Lipson. Evolving three-dimensional objects with a generative encoding inspired by developmental biology. In *Proc. of the European Conf. on Artificial Life*, pages 144–148, 2011.

[8] J. Clune, J.-B. Mouret, and H. Lipson. The evolutionary origins of modularity. *Proc. of the Royal Society B*, 280(20122863), 2013.

[9] J. Clune, K. O. Stanley, R. T. Pennock, and C. Ofria. On the performance of indirect encoding across the continuum of regularity. *IEEE Trans. on Evolutionary Computation*, 15(4):346–367, 2011.

[10] J. Gauci and K. O. Stanley. Generating large-scale neural networks through discovering geometric regularities. In *Proc. of the Genetic & Evolutionary Computation Conf.*, pages 997–1004. ACM, 2007.

[11] J. D. Hiller and H. Lipson. Multi-material topological optimization of structures and mechanisms. In *Proc. of the Genetic & Evolutionary Computation Conf.*, pages 1521–1528. ACM, 2009.

[12] J. D. Hiller and H. Lipson. Evolving amorphous robots. *Artificial Life XII*, pages 717–724, 2010.

[13] J. D. Hiller and H. Lipson. Automatic design and manufacture of soft robots. *IEEE Trans. on Robotics*, 28(2):457–466, 2012.

[14] J. D. Hiller and H. Lipson. Dynamic simulation of soft heterogeneous objects. *ArXiv:1212.2845*, 2012.

[15] S. Hirose and Y. Umetani. The development of soft gripper for the versatile robot hand. *Mechanism and Machine Theory*, 13(3):351–359, 1978.

[16] G. S. Hornby, H. Lipson, and J. B. Pollack. Generative representations for the automated design of modular physical robots. *IEEE Trans. on Robotics and Automation*, 19(4):703–719, 2003.

[17] G. S. Hornby and J. B. Pollack. Evolving L-systems to generate virtual creatures. *Computers & Graphics*, 25(6):1041–1048, 2001.

[18] M. Joachimczak and B. Wróbel. Co-evolution of morphology and control of soft-bodied multicellular animats. In *Proc. of the Genetic & Evolutionary Computation Conf.*, pages 561–568. ACM, 2012.

[19] M. Komosinski and A. Rotaru-Varga. Comparison of different genotype encodings for simulated three dimensional agents. *Artificial Life*, 7(4):395–418, 2001.

[20] S. Lee, J. Yosinski, K. Glette, H. Lipson, and J. Clune. Evolving gaits for physical robots with the HyperNEAT generative encoding: the benefits of simulation., 2013.

[21] J. Lehman and K. O. Stanley. Evolving a diversity of virtual creatures through novelty search and local competition. In *Proc. of the Genetic & Evolutionary Computation Conf.*, pages 211–218. ACM, 2011.

[22] H. Lipson and J. B. Pollack. Automatic design and manufacture of robotic lifeforms. *Nature*, 406(6799):974–978, 2000.

[23] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *ACM Siggraph*, volume 21, pages 163–169, 1987.

[24] J.-B. Mouret and J. Clune. An algorithm to create phenotype-fitness maps. *Proc. of the Artificial Life Conf.*, pages 593–594, 2012.

[25] S. Nolfi, D. Floreano, O. Miglino, and F. Mondada. How to evolve autonomous robots: Different approaches in evolutionary robotics. In *Artificial Life IV*, pages 190–197. Cambridge, MA: MIT Press, 1994.

[26] R. Pfeifer and J. C. Bongard. *How the body shapes the way we think: a new view of intelligence*. MIT press, 2006.

[27] J. Rieffel, F. Saunders, S. Nadimpalli, H. Zhou, S. Hassoun, J. Rife, and B. Trimmer. Evolving soft robotic locomotion in PhysX. In *Proc. of the Genetic & Evolutionary Computation Conf.*, pages 2499–2504. ACM, 2009.

[28] S. Sanan, J. Moidel, and C. G. Atkeson. A continuum approach to safe robots for physical human interaction. In *Int'l Symposium on Quality of Life Technology*, 2011.

[29] J. Secretan, N. Beato, D. B. D'Ambrosio, A. Rodriguez, A. Campbell, and K. O. Stanley. Picbreeder: evolving pictures collaboratively online. In *Proc. of the 26th SIGCHI Conf. on Human Factors in Computing Systems*, pages 1759–1768. ACM, 2008.

[30] K. Sims. Evolving virtual creatures. In *Proc. of the 21st Annual Conf. on Computer Graphics and Interactive Techniques*, pages 15–22. ACM, 1994.

[31] K. O. Stanley. Compositional pattern producing networks: A novel abstraction of development. *Genetic Programming and Evolvable Machines*, 8(2):131–162, 2007.

[32] K. O. Stanley, D. B. D'Ambrosio, and J. Gauci. A hypercube-based encoding for evolving large-scale neural networks. *Artificial Life*, 15(2):185–212, 2009.

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

[34] K. O. Stanley and R. Miikkulainen. A taxonomy for artificial embryogeny. *Artificial Life*, 9(2):93–130, 2003.

[35] D. Trivedi, C. Rahn, W. Kier, and I. Walker. Soft robotics: Biological inspiration, state of the art, and future research. *Applied Bionics and Biomechanics*, 5(3):99–117, 2008.

[36] P. Verbancsics and K. O. Stanley. Constraining connectivity to encourage modularity in HyperNEAT. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 1483–1490. ACM, 2011.

[37] M. Wall. Galib: A c++ library of genetic algorithm components. *Mechanical Engineering Department, Massachusetts Institute of Technology*, 87, 1996.

[38] B. G. Woolley and K. O. Stanley. On the deleterious effects of a priori objectives on evolution and representation. In *Proc. of the Genetic and Evolutionary Computation Conf.*, pages 957–964. ACM, 2011.