Evolving Amorphous Robots

Jonathan D. Hiller[1], Hod Lipson[1]

[1]Computational Synthesis Lab
Mechanical and Aerospace Engineering
Cornell University, Ithaca, NY, 14853
Hod.Lipson@cornell.edu

Abstract

Research in evolutionary robotics has traditionally been limited to morphologies comprising rigid and discrete components, such as links connected with rotational or linear joints and actuators. Here, we demonstrate the evolution of robots with continuous and amorphous morphologies composed of multiple materials. Actuation is accomplished by periodic volumetric expansion and contraction of one or more of these materials. The challenges of representing evolvable multi-material freeform shapes and evaluation (simulation) of the resulting soft bodies are discussed. Several genotypic representations are explored which use a level-set threshold to generate the material distribution in the phenotype. Soft body simulation of the robot is accomplished using a relaxation algorithm to model the dynamics of the resulting amorphous machines under the actuation material expansion, gravity forces, and non-linear ground friction. These results open the door to a new design space that more closely mimics the freeform, amorphous and continuous nature of biological systems.

Introduction

The field of evolutionary robotics has explored methods for generating interesting and functional robot morphologies (Nolfi and Floreano 2002). Ever since the early work in evolving virtual (Sims 1994) and physical (Lipson and Pollack 2000) creatures, many examples have been published of evolved walking, (Pollack, Lipson et al. 2001) running (Zykov, Bongard et al. 2004; Hornby, Takamura et al. 2005), and swimming (Ijspeert and Kodjabachian 1999) robots. These simulations all use rigid-body simulations of discrete components connected by rotational or linear joints. Many interesting biological forms of locomotion, however, are not modeled well by rigid links and joints – such as the earthworm (Quillin 1999) and the amoeba (Mast 1926). More recent work on morphogenetic robotics explores the development of more complex morphologies using many rigid links, but these bodies are still inherently discrete and relatively sparsely connected (Hornby, Lipson et al. 2001; Bongard 2003).

In this paper we focus on evolving fully amorphous soft robots. Material distributions take the place of discrete links, and volume changing materials replace discrete actuators. These material distributions are not constrained to any given topology or shape. This freedom removes fundamental constraints, thereby opening a vast new design space to explore.

Traditional computer aided design (CAD) tools are typically inappropriate for designing amorphous machines with continuous morphology and actuation. Such tools rely on feature-based modeling approaches that work well for well-defined geometric primitives made of a single material. However, the lack of constraints on the shape and material distribution of soft robots indicate that existing CAD programs would be ineffective in their ability to fully take advantage of the design space offered. Therefore, new higher level design tools are necessary to meet functional goals without geometric constraints.

As greater computing power becomes more readily available, design automation algorithms are becoming increasingly valuable for designing structures with freeform material distributions. Homogenization techniques (Bendsoe and Kikuchi 1988) are useful for designing single material structures such as 2D and 3D beams, and simple mechanisms(Nishiwaki, Frecker et al. 1998). However, homogenization techniques are limited in their ability to meet high level functional goals, such as specific beam deformations (Hiller and Lipson 2009) or locomotion. Here, we focus on the use of evolutionary algorithms to autonomously design locomoting amorphous soft robots.

We first briefly describe the field of soft robotics and the additive manufacturing technology that make amorphous robots possible. Next, we explore three representations that enable genetic algorithms to evolve functional three dimensional multi-material morphologies independent of topology. We then describe our soft body physics simulator used to evaluate potential solutions. Finally, the abilities of each representation are compared and several resulting amorphous, locomoting robots are shown for various scenarios.

Background

Soft Robotics

Robots are traditionally made of discrete rotary or linear actuators, connected by rigid links. This architecture is driven by the manufacturing technology used to physically build the robots and the methods used to simulate and control them. The kinematics of these machines can be deterministically modeled and used to perform useful functions such as path planning and collision avoidance.

A new paradigm in robotics has recently emerged, inspired largely by the robustness and resilience of biological systems. These "soft" robots trade deterministic control for probabilistic models, but gain robustness (Rieffel, Trimmer et al. 2008). While several actuation methods have been explored for soft robots such as shape memory alloy (SMA), pneumatics, (Rieffel, Saunders et al. 2009) electroactive polymers, (Bar-Cohen 2001) and jamming (Mozeika, Steltz et al. 2009), these all place constraints on the geometry and the ways in which internal forces are applied. Here we consider pure volumetrically actuated materials in order to more accurately mimic distributed actuation and avoid imposing undue constraints.

Freeform Fabrication

The new design space of soft robots is characterized by a nearly complete freedom over the spatial distribution of materials (Beaman, Marcus et al. 1997). Physically, this is realized by novel additive manufacturing technologies (also known as solid freeform fabrication, rapid prototyping or 3D printing). This technology is currently capable of autonomously fabricating multi-material 3D objects in any desired shape, with any internal material distribution (Malone, Rasa et al. 2004; Malone and Lipson 2007; Hiller and Lipson 2009; Objet 2010). Materials that can be co-fabricated include rigid plastics and soft rubbers.

A significant missing link in soft robots becoming ubiquitous is the ability to print volumetric actuators. Many examples are present in literature of additively manufactured robots with actuators added after fabrication. However, these are limited to traditional rotational or linear actuators, (Pollack, Lipson et al. 2001) which would severely limit the generality and methods of actuation of an amorphous robot. Here in simulation we explore an ideal volumetric actuator, in which a given material expands isometrically (equally in all dimensions). A useful analogy is that we will be evolving robots with materials of varying thermal expansion coefficients (CTE), then "actuating" the robot by globally or locally varying the "temperature". Thus materials with a simulated CTE of zero will not change volume, whereas materials with a non-zero CTE will swell or contract isometrically as the temperature changes.

In these experiments, the temperature is assumed to vary sinusoidally over time, and slowly enough that actuation across the entire structure occurs simultaneously without heat diffusion effects. The period and amplitude of this temperature variation determine how dynamic the movement of the robot is. More complex actuation patterns including evolved brains will be examined in the future.

Methods

In this section we address the two main challenges of evolving amorpheous soft robots. First, we explore continuous representations of 3D multi-material objects unconstrained by topology, with the goal of maximizing interesting shapes and evolvability while minimizing the number of variables to be evolved. Each continuous representation is rendered to discrete voxel-space for simulation, which allows any suitable resolution to be used for the simulation process in order to balance computational efficiency with accuracy. Second, we outline our soft-body physics engine used to efficiently simulate the dynamics of amorphous robots with non-linear large deformations, volume-changing materials, and friction.

Representations

There are many possible representations for three dimensional freeform shapes for an evolutionary algorithm. Most prior examples use primitives (Sims 1994) but these are not conducive to creating smooth freeform shapes. We use a level-set class of representations that create a four-dimensional landscape, which is then threshholded to create a three dimensional solid (Sethian and Wiegmann 2000; Wang, Wang et al. 2003). A convenient analogy is to view the genotype as specifying a 3D density field, to which a threshold is applied. All the volume at a higher density is instantiated as part of the solid, whereas the rest is interpreted as empty space.

The level-set concept is versatile and useful for evolving shapes for several reasons. First, there is complete freedom in the topology of the object. More importantly, a continuous evolution path between different topologies exists since a phenotype's topology is derived, not prescribed. Moreover, this representation allows multiple materials to be seamlessly interspersed throughout the volume. A density field for each material is generated. Then the boundary of the volume is determined by thresholding the sum of the density fields of each material at each location. The material with the highest density at each location within the lattice is instantiated at that location. Alternatively, mixtures of materials could be described by blending materials in ratios proportional to their respective density fields.

We explore three different representations that create 3D density fields: (a) The Discrete Cosine Transform (DCT) representation (Hiller and Lipson 2009), (b) the Compositional Pattern Producing Network (CPPN) representation (Stanley 2007), and (c) the Gaussian Mixtures representation (Pernkopf and Bouchaffra 2005). Each of these was chosen to create smooth shapes of multiple materials. Each representation is also open ended in that it has the ability to increase the complexity of the resulting objects at the expense of the number of evolved parameters.

Discrete Cosine Transform (DCT). The discrete cosine transform is a special case of the discrete Fourier transform, in which boundary conditions favorable to creating amorphous morphology shapes are enforced. In the DCT representation (Hiller and Lipson 2009) the phenotype is a 3D matrix of frequency amplitudes, ranging from -1 to 1. To convert each phenotype to a genotype, the inverse DCT is applied to each row of each dimension of this matrix, converting from the frequency domain to the spatial domain. Thus, each element in the frequency matrix scales a harmonic density field, where the number of modes in each of the three dimensions corresponds to its X, Y, and Z indices in the frequency matrix. A simple 1D example is shown in Figure 1. In this example, the 1D genotype matrix would be as follows:

$$\begin{bmatrix}0.5 & -0.2 & -0.6 & 0.5 & -1.2\end{bmatrix}$$

The first element of this matrix scales the fundamental harmonic, the second element scales the second harmonic, and so on. These weighted harmonic functions are then summed to create a density field, which is thresholded at zero. In the 1D case, this results in a “freeform” 1D line segment, as shown in red in Figure 1. By extension, a 3D matrix of frequency components results in a freeform 3D solid.

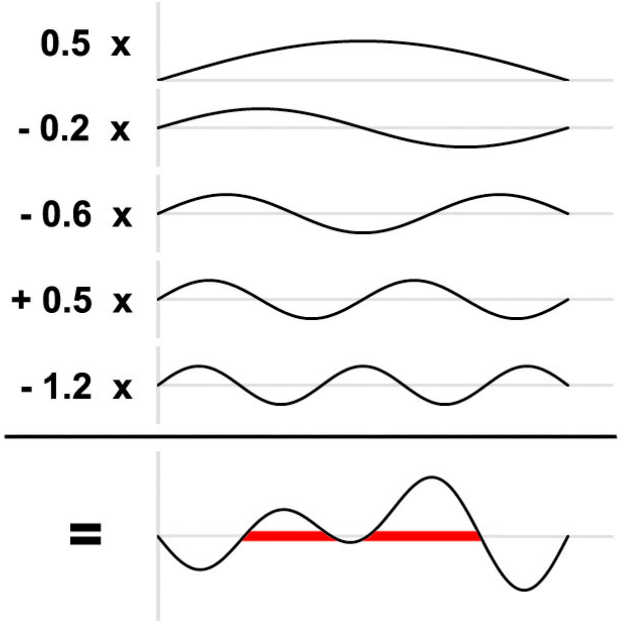


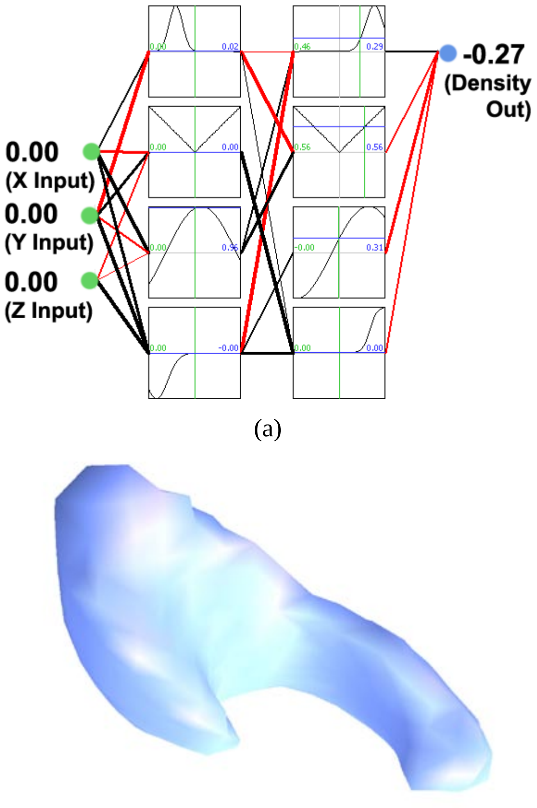
Figure 1: The inverse discrete Fourier transform representation sums weighted sinusoids, then thresholds them at zero as shown in this 1D example. The weights are the only evolved parameters.

The usefulness of the DCT representation for creating smooth, amorphous shapes is realized when the evolved frequency amplitude matrix is smaller than the rendered matrix of voxels in the spatial domain. Before the inverse DCT is applied, the frequency amplitude matrix of the genotype is simply padded with zeros to match the dimension of the number of voxels in the phenotype. Thus, smooth, freeform shapes are created.

When evolving freeform amorphous morphologies using the DCT representation, mutation involves making small changes (up to 5%) in amplitude of these frequency components. A mutation rate of 20% was used. The crossover operation randomly selects each frequency component from either parent to create offspring.

Compositional Pattern Producing Network (CPPN). Compositional Pattern Producing Networks (CPPNs) (Stanley 2007) have been demonstrated to be useful for evolving two dimensional density fields (often interpreted as grey-scale images). Here, we introduce the third dimension to produce 3D density fields to threshold into amorphous morphologies. CPPNs are similar in concept to an artificial neural network (ANN), except that more geometrically-useful transfer functions are used instead of just sigmoids. A network of nodes (each containing a function) are connected by weighted paths. In order to create 3D amorphous morphologies, three coordinates (X, Y, and Z) that represent the position of a point in 3D space are used as inputs. The network has a single output, which represents the resulting density at that point. By sweeping through X, Y, and Z, the full 3D density field is obtained

Unlike ANNs, however, a variety of activation functions are used in a CPPN. Activation functions used here include traditional sigmoids and Gaussians, as well as sinusoids and the absolute value function for inducing repetition and symmetry, respectively. For each node (function), several parameters were evolved. These include the function type, offsets, and scaling. Additionally, a complexity measure was implemented to control minimum feature sizes, such that features were not being lost at a sub-voxel scale. Weights between nodes were also subject to evolution. An example CPPN and the resulting geometry is shown in Figure 2.



(a)

(b)

Figure 2: The Compositional Pattern Producing Network (CPPN) representation evolves a network of functions with three inputs (X, Y, and Z) and one output, which is the density at that location. The node functions and connecting weights (negative shown red, positive shown black) are evolved. After sweeping the inputs and thresholding, the network (a) produces a 3D freeform shape (b).

A CPPN has many evolvable parameters. Given a network that is m layers deep with n nodes per layer, there are a total of m×n nodes. Each node has an assigned activation function and three parameters that describe it (offset and scaling along the X axis, and scaling along the Y axis). There are also as many as $(m-1)\times n^2$ real-value weighted connections between the nodes, which can be either active or inactive. All these parameters are eligible for small changes upon mutation. However, the mutation rate is chosen such that on average only one of these values (in total) is adjusted. In the crossover operation, a rectangular "region" of nodes is selected from one parent, and the rest of the nodes are taken from the second parent. This region is chosen such that all nodes are equally likely to be in the selected region, not favoring the center nodes.

Gaussian Mixtures (GMX). The Gaussian mixtures representation also relies on the density field analogy of the level-set method common to all these representations. In the GMX representation, the density field is initialized with zero density everywhere. Then, points of density with Gaussian falloff are added within the spatial envelope. These points can have either positive or negative weights, which add or subtract from the density respectively. If only one Gaussian point was used, the resulting thresholded solids would always be spheres. However, with a relatively small number of Gaussian Mixtures, interesting and complex shapes and topologies can result. A simple 2D example with equal size and equally weighted Gaussian points is shown in Figure 3.

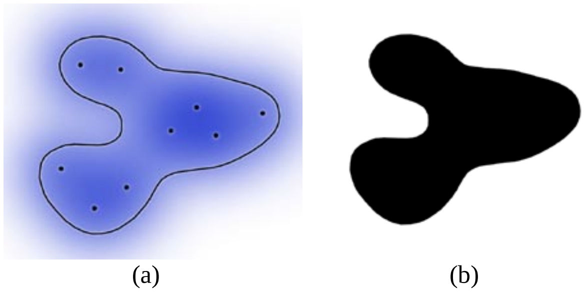


Figure 3: In the Gaussian Mixtures representation, the locations and intensities of Gaussian distributions are evolved. In this simple 2D example, nine Gaussian point locations with equal sizes and weights (a) are threshholded to create a freeform 2D shape (b).

Mutating the Gaussian Mixtures representation involves making small changes to the location, density, and falloff (radius) of a Gaussian point, and occasionally adding or removing points. A mutation rate of 20% was used. Crossing over two individuals is accomplished by initializing a random plane that intersects the volume of the workspace. Points from one side of the plane are taken from one parent, while points from the other side of the plane are taken from the second parent. Here we used only spherical distributions though general Gaussians could be used as well by representing each distribution using a covariance matrix.

Soft Body Simulation

In order to efficiently evaluate amorphous soft robot morphologies with volumetric actuation, we developed a soft body simulator ab initio in C++. The main features of our simulation are:

1) **Speed:** With thousands of time steps per evaluation, and thousands of evaluations per evolutionary run, the feasibility of evolving amorphous robots depends on having an efficient simulation.
2) **Dynamics:** Full dynamics modeling with variable damping allows for realistic, 2nd order momentum effects in all translational and rotational degrees of freedom.
3) **Large deformation:** Shapes can be bent and twisted far past any linear small angle approximations without revoxelizing.
4) **Multi-material:** Any number of materials can be combined in any internal material distribution, each with varying stiffnesses and densities.
5) **Friction:** Nonlinear friction is incorporated with a static/dynamic friction model.
6) **Collision detection and handling:** Self intersection is calculated and enforced. With large deformation comes the need to avoid an object penetrating itself.

When a continuous amorphous robot object is imported into the simulation, it is first voxelized at an appropriate resolution. These voxels are then simulated according to the appropriate statics and dynamics, and the continuous mesh is drawn according to the deformation of the nearest voxel (Alec and Doug 2007). At each time step, the total force on each voxel is calculated. Then, the momentum (P) of each voxel is updated according to the length of the time step (Δt) and the total force.

$$P_t = P_{t-1} + \sum F \times \Delta t$$

Linear damping was modeled, which is consistent with the internal damping of most bulk materials. The loss factor (η) is normalized by the length of time step and determines how much energy (in the form of momentum) is lost at each time step.

$$P_{new} = P_{old} \times \eta$$

Finally, the momentum is numerically integrated to get the change in position (ΔX) of the voxel. The positions of each voxel are synchronously updated, and the process repeats.

$$\Delta X = \frac{P \times \Delta t}{m}$$

Although the equations above illustrate the translational degrees of freedom, the equivalent equations are used to model the rotational degrees of freedom. An example of the freeform, large-displacement nature of this soft-body simulation is shown in Figure 4.

The interaction between individual voxels is modeled by a standard flexible beam model. This allows all 6 relative forces and moments to be calculated based on the relative 3D position and rotation of the two voxels. For computational efficiency, each element is transformed to point in the positive X direction before the reactions are calculated. The reaction forces and moments then undergo the inverse transform to put them back in the reference frame of the actual element. When considering two voxels of differing material properties (such as stiffness), the bond is assumed to have the composite stiffness of the two materials connected in series.

Choosing the optimal time step is critical to an efficient simulation. If the time step is too small, computation time is wasted with the extra time steps. However, a time step that is too large will result in diverging instability within the simulation. Calculating the optimal time step for an arbitrary geometry with varying stiffness in the material and non-linear interactions such as collisions and friction is non-trivial. To address this we experimentally determine the optimal time step upon importing an object to the simulation. This involves a series of short simulation runs, with steadily decreasing time steps. Since divergence is exponential, very few time steps are needed to determine if a given time step increment is unstable. Thus, the first simulation without a significant increase in strain energy by 100 time steps is assumed stable, and a safety factor of 5% is incorporated. For efficiency, very coarse steps are taken at first (one order of magnitude apart), then a second finer pass determines the optimal time step at a higher resolution.

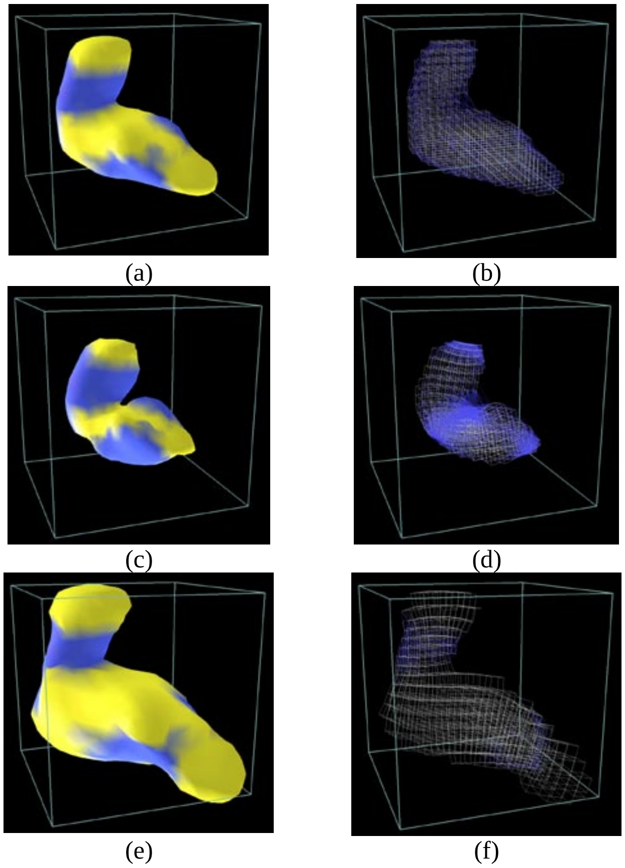


Figure 4: A randomly generated 3D object (a) is imported into the soft-body simulation, which voxelizes the object at an appropriate level of detail. (b) Potential self-intersection collisions are shown as blue lines. As the temperature varies, the volume of the yellow material shrinks (c & d) and swells (e & f) accordingly. The volume of the blue material remains constant.

Calculating self intersection is necessary in a soft-body simulation since the objects deform significantly. However, calculating collisions for a large number of lattice points is computationally intensive. This has an order of n^2 complexity, where the number of lattice points n can easily be in the thousands. This would dominate the computational time of the physical simulation itself, which is order n complexity. To address this, we made use of several useful simplifications. First, only the voxels on the exterior of the object need be considered for collisions. Second, an intermediate list of possible collisions between bonds can be maintained. Initially, each pair of voxels within an absolute distance 2.5 voxels but which are not touching within two links in the lattice are added to the list. Then, at each time step only the possible contact bonds on this list are considered. The list is periodically regenerated when the absolute displacement of any voxel in the lattice is enough to touch a voxel not on its list of possible interactions.

Several parameters of the simulation were chosen to be of interest for exploring further. The first is the level of dynamic response. This term refers to the importance of the momentum term of the material. An object with a high level of dynamic response could be a very dense, soft, rubbery object actuated near resonance, where movements can be significantly out of phase with the actuation. Conversely, an object with a low level of dynamic response would be light and stiff (or actuated very slowly), such that the static movement dominates the momentum terms. Several combinations of static and dynamic friction were also explored, ranging from realistic values to exaggerated stick-slip scenarios. For the bulk of experiments, the coefficient of dynamic friction was 0.3 and the coefficient of static friction was 1.0.

Each evaluation of an amorphous machine was broken into two segments. The relaxation segment settles the object under gravity and friction, allowing it to come to rest in a neutral position. In the movement segment, temperature oscillations begin. After 10 complete temperature cycles, the magnitude of change in position of the center of mass during the movement segment is returned as the fitness for a given individual.

Evolution parameters and performance

The solutions presented here were each evolved on a single quad-core desktop computer. Each solution was voxelized into a 20×20×20 workspace, which provided suitably accurate resolution while remaining computationally feasible. At a rate of approximately 3-15 seconds per evaluation (depending on number of instantiated voxels), 20,000 evaluations in a 24 hour day was typical. Deterministic crowding selection was used (Mahfoud 1996), in which an offspring replaces its most similar parent if it outperforms it. Small population sizes work well with this crowding method, so a population size of 20 was chosen for all experiments. The mutation rate was different for each representation as detailed above.

Results

The evolved behaviors of the amorphous robots generally took advantage of a combination of dynamics and non-linear friction to make forward progress. Two modes of movement in the desired direction were generally observed: Several robots made significant progress by maximizing the distance traveled as they fell and flopped over. This movement was often aided by the actuation cycles gradually tipping the robot over, but this method of movement does not count as true locomotion because it cannot be sustained over an indefinite distance. The more successful mode of movement involved scooting, in which the robots expanded and contracted in specific ways, making and breaking contact friction selectively to make forward progress. Several observed solutions made use of a combination of flopping and scooting.

Two material results

In the first experiments, a palette of two equal stiffness and density materials was used. The first material (shown in blue) had a CTE of zero, signifying that it was not actuated. The second material (shown in yellow) had an arbitrary CTE of 0.01. The temperature was varied sinusoidally globally with an amplitude of 30 degrees, leading to a ±30% change in volume of the actuated material. The period of oscillation was 500 time steps.

Comparison of Representations

The three representations under consideration were all run three times for a total of nine evolutionary runs. Figure 6 displays the average and standard error of the best solution of each of the three runs. The GMX representation outperformed the other representations consistently. This may be a result of locality that preserves geometric novelties in the crossover process and can make small changes to specific areas of the robot through mutation.

The DCT representation fell behind and had a very large standard error, which means that the genetic algorithm was not able to consistently find good solutions. This is likely because each mutation in the genome has a global effect across the entire structure, a characteristic that couples the mutations and prohibits small, subtle changes.

The CPPN representation, as implemented, was not well suited to evolving freeform amorphous morphologies. Mutations drove the solution toward filling the entire workspace with material, a trend that significantly slowed the simulation down (since more elements needed to be simulated) and also led to fewer interesting geometries. Resulting amorphous robots generated by each representation are shown in Figure 5.

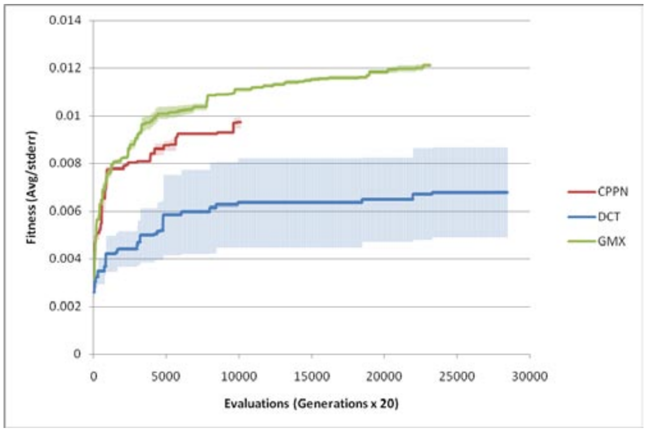


Figure 6: Three independent runs were completed for each of the three representations. The average and standard error of the three best solutions are plotted for each. The GMX representation outperformed the others.

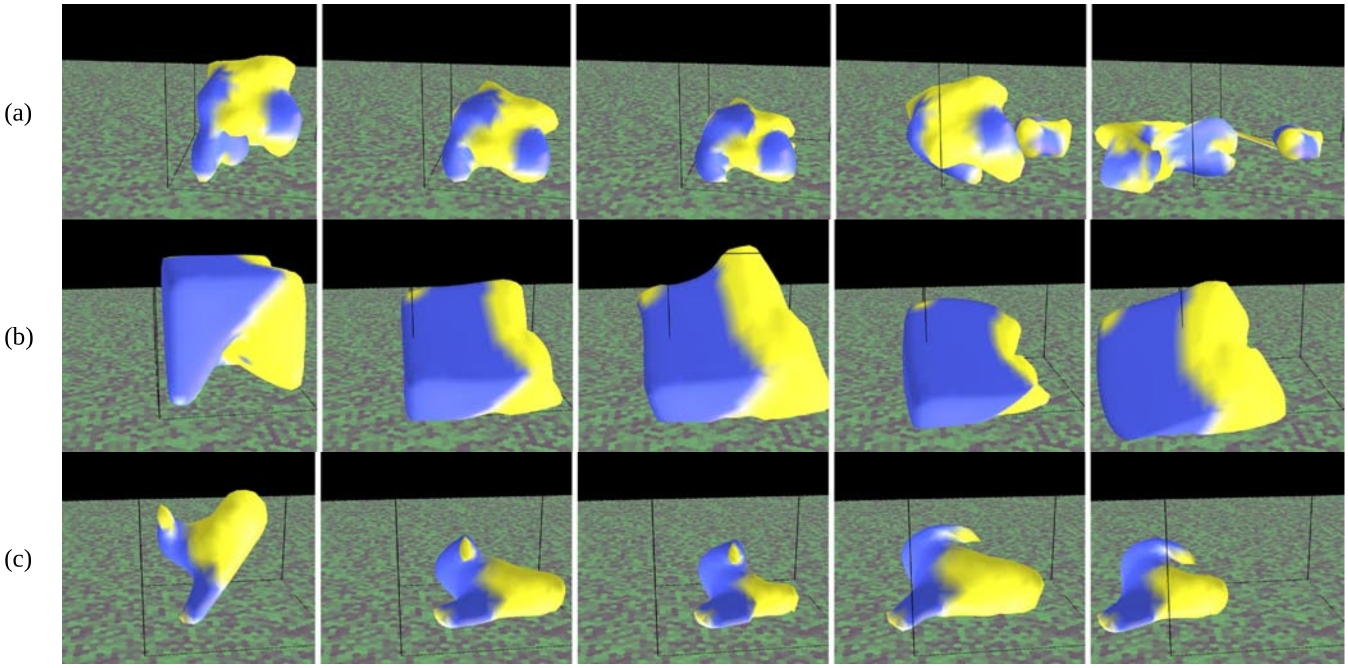


Figure 5: Evolved robots for the DCT (a), CPPN (b), and GMX (c) representations demonstrate successful locomotion. The blue material is passive, while the yellow material changes volume sinusoidally. The first frames for each show the initialized shape, the second frames show the settled result under gravity, and the following frames are snapshots of its motion. Direction of motion is to the left.

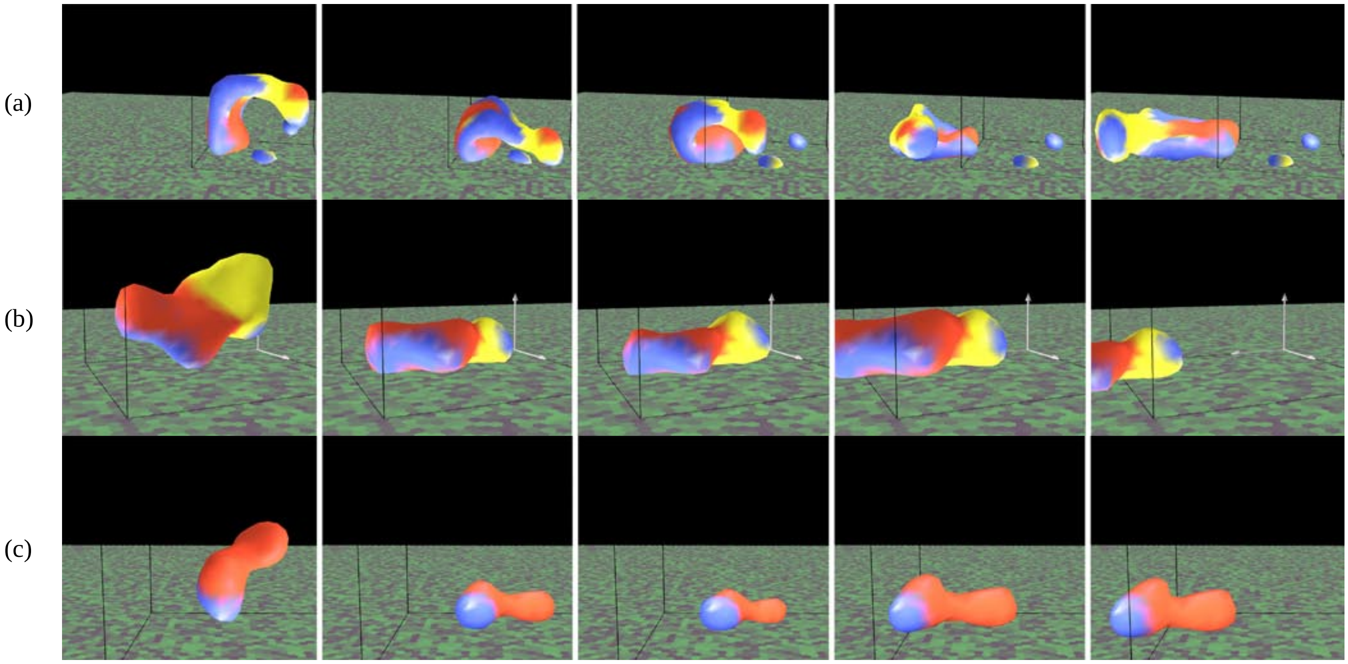


Figure 7: Evolved amorphous morphologies showing flopping (a), scooting (b), and dynamic bouncing (c) behaviors. The yellow and red materials are both sinusoidally actuated, but 90 degrees out of phase. Direction of motion is to the left.

Three material results

A second volumetrically actuated material was introduced to explore the possibilities enabled by multiple actuation modes. The second actuator material was simulated the same stiffness and density as the others, but with 90 degree phase lag in actuation. It was hypothesized that this would enable different modes of locomotion, such as continuous rolling or more interesting gaits. However, only the more primitive locomotion modes of flopping (Figure 7a) and scooting (Figure 7b) were observed.

Dynamic response. By varying the actuation speed of the temperature fluctuations and the internal material damping, the importance of momentum effects in the amorphous morphologies can be adjusted. The best solution of the dynamic runs ended up using only one actuator material, as shown in Figure 7c. Based on the size and mass of the optimized object, however, the dynamics were strongly exploited to bounce forward.

Friction. Different parameters for friction bias the solution towards different modes of locomotion. Experiments were run with very low dynamic friction and high static friction (0.1/5.0) and with dynamic and static friction values that were very close (0.4/0.5). The solutions with very high static friction tended to exhibit flopping/rolling over behavior, such as in Figure 7a, since the force to overcome static friction was extremely high. However, solutions with moderate static and dynamic friction tended towards the scooting locomotion, such as in Figure 7b.

Conclusions

We have demonstrated that evolutionary algorithms are suitable for designing the freeform material distribution of locomoting amorphous robots, which would be a difficult task to perform in traditional CAD software. This opens the door to a new design space of soft robotics, where the functionality of the robot is determined by the material distribution, not by rigid links. Sensing, actuation, and computation can all be distributed, potentially making the design of these robots even more difficult without the aid of design automation methods. Thus, with the exponentially expanding design space of robotics enabled by additive manufacturing of multiple materials, genetic algorithms and other design automation methods will play an increasingly important role in designing robots to directly meet high level functional goals.

Acknowledgements

This work was supported in part by a National Science Foundation Graduate Research Fellowship and NSF Creative-IT grant 0757478.

References

Alec, R. R. and L. J. Doug (2007). FastLSM: fast lattice shape matching for robust real-time deformation. ACM SIGGRAPH 2007 papers. San Diego, California, ACM.

Bar-Cohen, Y. (2001). Electroactive Polymer (EAP) Actuators as Artificial Muscles - Reality, Potential and Challenges, SPIE Press.

Beaman, J. J., H. L. Marcus, et al. (1997). Solid Freeform Fabrication: A New Direction in Manufacturing, Kluwer Academic Publishers.

Bendsoe, M. P. and N. Kikuchi (1988). "Generating optimal topologies in structural design using a homogenization method." Comput. Methods Appl. Mech. Eng. **71**(2): 197-224.

Bongard, J. (2003). Incremental Approaches to the Combined Evolution of a Robot’s Body and Brain. Zurich, University of Zurich. **Ph.D.:** 193.

Hiller, J. and H. Lipson (2009). "Design and Analysis of Digital Materials for Physical 3D Voxel Printing." Rapid Prototyping Journal **15**(2): 137-149.

Hiller, J. and H. Lipson (2009). Multi Material Topological Optimization of Structures and Mechanism. Genetic and Evolutionary Computation Conference. Montreal, QC.

Hornby, G. S., H. Lipson, et al. (2001). Evolution of generative design systems for modular physical robots. Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on.

Hornby, G. S., S. Takamura, et al. (2005). "Autonomous evolution of dynamic gaits with two quadruped robots." Robotics, IEEE Transactions on **21**(3): 402-410.

Ijspeert, A. J. and J. r. m. Kodjabachian (1999). "Evolution and Development of a Central Pattern Generator for the Swimming of a Lamprey." Artificial Life **5**(3): 247-269.

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

Mahfoud, S. (1996). "Niching methods for genetic algorithms."

Malone, E. and H. Lipson (2007). "Fab@Home: the personal desktop fabricator kit." Rapid Prototyping Journal **13**(4): 245-255.

Malone, E., K. Rasa, et al. (2004). "Freeform fabrication of zinc-air batteries and electromechanical assemblies." Rapid Prototyping Journal **10**(1): 58-69.

Mast, S. O. (1926). "Structure, movement, locomotion, and stimulation in amoeba." Journal of Morphology **41**(2): 347-425.

Mozeika, A., E. Steltz, et al. (2009). The First Steps of a Robot Based on Jamming Skin Enabled Locomotion. International Conference on Intelligent Robots and Systems (IROS). St. Louis, USA**:** 408-409.

Nishiwaki, S., M. I. Frecker, et al. (1998). "Topology optimization of compliant mechanisms using the homogenization method." International Journal for Numerical Methods in Engineering **42**(3): 535-559.

Nolfi, S. and D. Floreano (2002). "Synthesis of autonomous robots through evolution." Trends in Cognitive Sciences **6**(1): 31-37.

Objet. (2010). "Objet Geometries inc." Retrieved January 24, 2010, from http://www.objet.com.

Pernkopf, F. and D. Bouchaffra (2005). "Genetic-based EM algorithm for learning Gaussian mixture models." Pattern Analysis and Machine Intelligence, IEEE Transactions on **27**(8): 1344-1348.

Pollack, J. B., H. Lipson, et al. (2001). "Three Generations of Automatically Designed Robots." Artificial Life **7**(3): 215-223.

Quillin, K. J. (1999). "Kinematic scaling of locomotion by hydrostatic animals: ontogeny of peristaltic crawling by the earthworm lumbricus terrestris." J Exp Biol **202**(6): 661-674.

Rieffel, J., F. Saunders, et al. (2009). Evolving soft robotic locomotion in PhysX. Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers. Montreal, Quebec, Canada, ACM.

Rieffel, J., B. Trimmer, et al. (2008). Mechanism as Mind: What Tensegrities and Caterpillars Can Teach Us about Soft Robotics. Artificial Life (ALIFE) XI. Winchester, UK**:** 506-512.

Sethian, J. A. and A. Wiegmann (2000). "Structural Boundary Design via Level Set and Immersed Interface Methods." Journal of Computational Physics **163**: 489-528.

Sims, K. (1994). Evolving virtual creatures. SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques. Orlando, Florida, ACM**:** 15-22.

Stanley, K. (2007). "Compositional pattern producing networks: A novel abstraction of development." Genetic Programming and Evolvable Machines **8**(2): 131-162.

Wang, M. Y., X. Wang, et al. (2003). "A level set method for structural topology optimization." Computer Methods in Applied Mechanics and Engineering **192**(1-2): 227-246.

Zykov, V., J. Bongard, et al. (2004). Evolving Dynamic Gaits on a Physical Robot. GECCO '04.