Automatic Design and Manufacture of Soft Robots

Jonathan Hiller and Hod Lipson, *Member, IEEE*

***Abstract*—We present the automated design and manufacture of static and locomotion objects in which functionality is obtained purely by the unconstrained 3-D distribution of materials. Recent advances in multimaterial fabrication techniques enable continuous shapes to be fabricated with unprecedented fidelity unhindered by spatial constraints and homogeneous materials. We address the challenges of exploitation of the freedom of this vast new design space using evolutionary algorithms. We first show a set of cantilever beams automatically designed to deflect in arbitrary static profiles using hard and soft materials. These beams were automatically fabricated, and their physical performance was confirmed within 0.5–7.6% accuracy. We then demonstrate the automatic design of freeform soft robots for forward locomotion using soft volumetrically expanding actuator materials. One robot was fabricated automatically and assembled, and its performance was confirmed with 15% error. We suggest that this approach to design automation opens the door to leveraging the full potential of the freeform multimaterial design space to generate novel mechanisms and deformable robots.**

***Index Terms*—Design automation, evolutionary robotics, soft robotics.**

Fig. 1. Composite image shows a freeform soft robot locomoting from left to right. Actuation is provided by the orange material periodically varying in volume by 20%.

I. Introduction

In this paper, we focus on the automated design and fabrication of fully amorphous structures and soft robots. In soft robotics, discrete links and joints are replaced by continuous hard and soft material distributions. Single-degree-of-freedom (DOF) actuators are replaced by distributed volume changing materials. This emerging field is enabled by a combination of new materials and fabrication processes, new actuation techniques and new simulation algorithms that can efficiently predict the complex dynamics of soft materials.

The field of soft robotics is a relatively new area of research compared with traditional rigid-link robotics. Soft robots generally trade precision and deterministic control for bioinspired compliance and physical robustness. Such soft robots have potential for tasks in highly unstructured or dangerous environments, such as a collapsed building or situations with a high risk of a shifting environment or unexpected collisions. The inherent compliance of soft robots also uniquely suits them to tasks that involve contact with humans, animals, or delicate objects. In all of the aforementioned examples, it is expected that soft robots can be constructed to fulfill these tasks at a small fraction of the cost needed for an equivalent rigid robot.

The relatively unconstrained fabrication methods and material distributions of soft robots give rise to a large design space of form and function that is difficult to explore using traditional robot design paradigms. Manual computer-aided design (CAD) methods are well suited for designing individual components subject to the manufacturing constraints that are imposed by traditional fabrication methods, such as machining or casting. However, these complicated manufacturing constraints severely limit design automation routines, limiting their utility in traditional robot design to merely design optimization. Additionally, existing CAD tools are generally not well suited for a human designer to create the amorphous shapes and material distributions needed for soft robots. The complete lack of specialized geometric constraints both necessitates and enables a design automation algorithm to be well suited for such a design problem.

Design automation algorithms will become increasingly useful for designing and optimizing structures with freeform material distributions. Ideally, a designer would only need to input the functional goals and constraints of the desired object, and an optimal material blueprint would be automatically generated. As computing power continues to increase, the ability of algorithms to formulate and optimize largely unconstrained systems will outpace the ability of the human mind to do so. Early research in iterative mechanical design algorithms, such as homogenization techniques [1], demonstrated the proficiency of computers to design near-optimal single-material structures, such as 2-D and 3-D beams. However, these iterative homogenization techniques are limited in their ability to meet high-level functional goals, such as specifying a specific deformed shape [2] or locomotion. For this reason, here we use evolutionary algorithms

Manuscript received February 15, 2011; revised July 7, 2011; accepted October 12, 2011. Date of publication December 7, 2011; date of current version April 9, 2012. This paper was recommended for publication by Associate Editor K. Hosoda and Editor G. Oriolo upon evaluation of the reviewers' comments. This work was supported by a National Science Foundation (NSF) Graduate Research Fellowship and NSF Creative-IT Grant 0757478.

The authors are with the Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853 USA (e-mail: jdh74@cornell.edu; hod.lipson@cornell.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2011.2172702

to explore the design space in a more stochastic process that allows nondirect solutions to be obtained.

In previous work, we presented preliminary results of an evolutionary design automation algorithm as applied to simulated freeform multimaterial beams and amorphous robots [2], [3] and provide a detailed description of the soft-body simulation engine [4]. Here, were build on these results and demonstrate automatic design and *manufacturing* of static freeform beams and dynamic locomoting soft robots (see Fig. 1). The previously untested results of the design automation algorithm were fabricated and verified in the real world, and the results were compared favorably with the predicted performance.

The specific scenarios that are explored here are simply representative of the utility of the evolutionary design automation algorithm to meet high-level functional goals. Thus, such an algorithm can be used not just to design locomoting soft-robot morphologies but robots to perform other high-level tasks that could involve sensory input, such as overcoming obstacles, fitting through small holes, reacting to environmental changes, or seeking prey as well. Locomotion is simply a practical instance of this design automation method.

A. Soft Robotics

Robots traditionally are composed of rigid links that are connected by discrete single DOF rotary or linear actuators. Such robots can be very precise and are invaluable and ubiquitous in structured, well-known environments, such as industrial assembly lines. Although this precision is desirable in many cases, the downside is a lack of robustness. In order to avoid damage, the kinematics of these machines must be modeled deterministically and then carefully applied to do path planning and avoid collisions.

However, in uncertain and potentially harsh environments, the complexity, precision, and fragility of traditional robots can be a hindrance. Although robots have long been inspired by various aspects of biological systems, a new paradigm in robotics has recently emerged to replicate their robustness and resilience. These "soft" robots trade deterministic control for probabilistic models, but gain robustness [5]. Many varieties of soft actuators have been demonstrated, but several specific actuation methods have been explored that show promise for soft robots. These include jamming [6], electroactive polymers [7], pneumatics [8], and shape memory alloy (SMA) [9], but all place constraints on how the internal locomotion forces can be applied, as well as the resulting geometry. A summary of the current research in deformable soft robots is given in Table I.

Here, we propose incorporating an isometric volume-changing material for truly distributed actuation without geometric constraints. This allows complete freedom to the design automation algorithm over the method and magnitude of locomotion is based purely on material distributions. Thus, the function of an evolved soft robot is inextricably paired with its morphology without imposing additional forces at arbitrary locations. Since the complexity of how locomotion is achieved is embedded in the shape, a very simple control scheme can be used. Here, we simply impose a global sinusoidal variation in volume of the designated actuation material.

B. Freeform Additive Manufacturing

In order to automatically manufacture amorphous soft robots, new technology must be developed with the capability to fabricate soft morphologies and actuation. Additive manufacturing methods (which is also known as rapid prototyping or 3-D printing) allow for 3-D objects to be fabricated with minimal geometric constraints. These processes generally slice the target geometry into 2-D layers, which are then sequentially fabricated with finite thickness to build up a 3-D object. Initially, these manufacturing techniques were limited to creating objects of a single homogeneous material, such as steel, resin, or plastic [12]. However, multimaterial 3-D objects are now possible, with the ability to create arbitrary internal material distributions. Materials that can be cofabricated include rigid plastics and soft rubbers with as much as three orders of magnitude difference in stiffness using inkjet technology [15]. Progress has been made toward automated fabrication of functional robots with primarily soft materials [13] using extrusion techniques as well [14].

Additive manufacturing has been used to facilitate the autonomous construction of traditional robots using rigid limbs and rotational joints [16]. However, only the structural parts are fabricated automatically and the actuation is then added afterward by installing motors, batteries, and control circuitry [17]. Such robots exist within the traditional discrete paradigm of robots discussed earlier. This severely limits the application of existing methods toward the fabrication of an amorphously actuated robot. Attempts to incorporate an electroactive polymer actuator material directly into the fabrication process have been demonstrated [13], but the actuator performance is not yet on par with comparable discrete actuation methods.

C. Topology Optimization

General shape-optimization techniques seek to optimize the parameters or shape of an existing structure. In contrast, topology optimization is not bound to a pre-existing structure, shape, or topology. This allows it more flexibility to adapt to fundamentally different solutions. Topological optimization has been well explored for maximizing rigidity of 3-D elastic structures, such as cantilever beams under a predefined loading scenario [18], [19]. However, these utilize a single homogeneous material and optimize only a single parameter, such as stiffness per weight of the global structure, in contrast with the multimaterial scenario with higher level functional goals that are demonstrated here.

A homogenization technique to solve this class of topology optimization problems has been demonstrated in [1]. This iterative process alternately performs a physical simulation and then varies local stiffness based on the results of the previous simulation. Pixels that are located in the regions of high strain energy are strengthened, whereas pixels in the regions of low strain energy are gradually eliminated. The structure as a whole is updated subject to global constraints on the total volume and minimizing the total strain energy under load. Variations on

TABLE I
SUMMARY OF CURRENTLY PUBLISHED DEFORMABLE SOFT ROBOTS

Author/Citation	*Materials*	*Actuation*	*Design method*	*Fabrication*	*Comments/Description*
B. Trimmer [10]	Silicone rubber	Shape memory alloy	Manual	2D cast and manually rolled into cylinder	Bio-inspired design from Manduca Sexta caterpillar.
H. Jaeger/iRobot [6]	Silicone rubber	Pneumatic/jamming	Manual	Lost wax cast and manual assembly	Regular 20-sided icosahedron shape.
S. Hiria [9]	Rubber/spring steel	Shape memory alloy	Manual	Traditional Manual	Regular cylinder and sphere shapes. Able to jump.
G. Whitesides [11]	Silicone rubber	Pneumatic	Manual	Cast and manual assembly	Potential for locomotion, but not fully realized.
Hiller & Lipson [3]	(Simulation only)	(Simulation only)	Fully automatic	(Simulation only)	Design only. No physical robot constructed.
[Results presented here]	**Silicone foam rubber**	**Environment pressure modulation**	**Fully automatic**	**Automated layer fabrication with indexed manual assembly**	**Unconstrained freeform shapes and distributed actuation. Untethered.**

*Nonmobile robotic systems with soft actuators but rigid kinematics are excluded.

this method have yielded results that maximize deflection or compliance of an overall object [21] or microstructure [22], or even utilize two materials to emulate an actuated structure [20]. However, homogenization techniques depend on updating each location directly based on a global parameter, which generally limits them to optimizing overall deflection or force. Upon incorporation of competing objectives or high-level functional goals into the optimization as presented in this paper, homogenization approaches become unwieldy or even completely impractical.

Evolutionary algorithms are often presented as an alternative to homogenization techniques. Evolutionary algorithms are much more general and can accommodate competing objectives and high level goals, but at the expense of much more processing time. The success of evolutionary algorithms in topology optimization depends strongly on the encoding that is used to represent the physical object. A direct encoding of the genome is often used in which every individual pixel is stored and manipulated by the search algorithm [23], [24]. This method scales very poorly to large structures because the number of pixels scales with the square of the resolution, which directly corresponds to a rapidly increasing size of the genome. In 3-D, this problem is exaggerated still further because the number of voxels (3-D pixels) scales with the cube of the resolution. Thus, efficient representations are key to enabling evolutionary algorithms to overcome the functional complexity limitations of homogenization, especially in three dimensions, as described in the following section.

D. Representation for Amorphous Shapes

It is desirable for the representation that is used within an evolutionary algorithm to define a desired subspace of sensible objects using a minimum number of parameters [25]. Various graph structures, generative encodings, and constrained bit-wise encodings that have been proposed to address these challenges have been summarized in [26]. Most examples of evolutionary algorithms within the specific field of evolutionary robotics use a collection of 3-D primitives [27] to represent a physical robot. However, these are not directly conducive to creating smooth freeform shapes.

The level-set concept is versatile and useful for evolving freeform shapes because there is no prescribed topology of the object. In the level set method, an n-dimensional height map is thresholded into an n-dimensional set of regions. For instance, a 2-D height map (which can be visualized as a 3-D surface or terrain map) can be thresholded (or chopped) at a specific height to generate the 2-D "topo" regions (see Fig. 2). If the height map is smooth, then smooth 2-D regions are generated. In order to generate smooth 3-D regions, a 3-D density field is generated, which is then thresholded to create a 3-D solid [28], [29].

It is possible to create interesting, smooth 3-D landscapes for the level-set method with a very reasonable number of evolved parameters. We specifically explore the Gaussian mixtures representation [30]. In this method, a list of Gaussian points within the 3-D workspace is maintained. Each point has an associated density and the Gaussian falloff range. To generate the 3-D density field, the relative density at each location in the workspace is calculated based on the linear sum of contributions from all Gaussian points. An illustration of the Gaussian mixtures in two dimensions is shown in Fig. 2. A single point [see Fig. 2(a)] results in a circular section. Two points [see Fig. 2(b)], i.e., one positive and one negative, creates a non-uniform shape. Twenty randomly generated points [see Fig. 2(c)] generate interesting smooth 2-D shapes. For the results that are presented here, this same concept was used in 3-D. Here, we used only uniform 3-D falloff for the Gaussian points, although representing each distribution using a covariance matrix would lead to more complex shapes with more free parameters. This choice of representation implies that solutions will be limited to smooth, freeform shapes rather than the more traditional 3-D primitives that are used in the evolution of robot morphologies. A comparison of the performance of the Gaussian mixtures representation with other freeform 3-D representations may be found in [3].

In order to extend the Gaussian mixtures representation to multiple materials, a material index for each Gaussian point is maintained. For each possible material in the simulation, the corresponding Gaussian points are rendered to a separate density field. Then, the final heterogeneous object is determined by the relative density of each material at each location. The total density of all materials is thresholded to obtain the shape. Within the shape, the material with the highest density at each location is then instantiated.

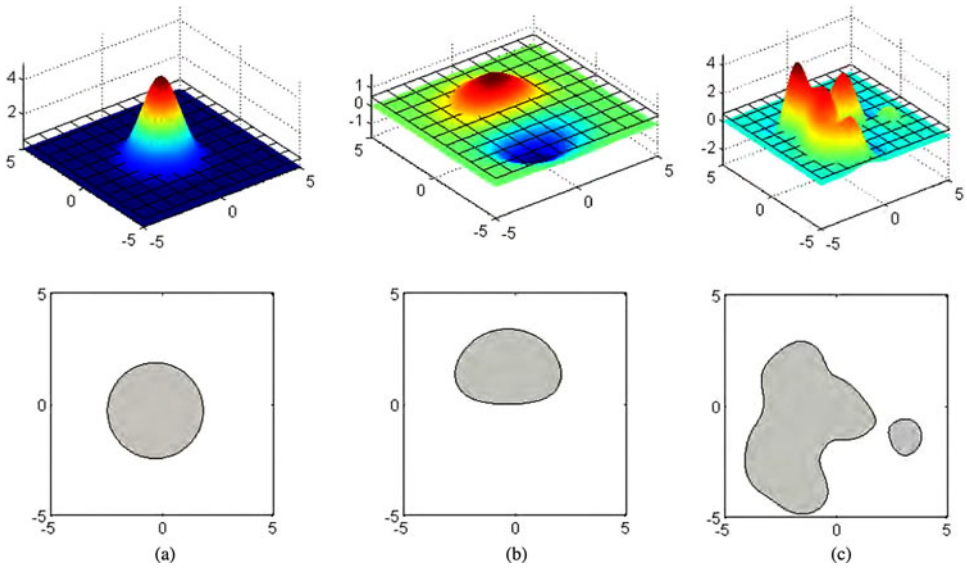

Fig. 2. Level set method of the Gaussian mixtures is illustrated in two dimensions. A single Gaussian density point thresholded at an arbitrary level yields a circular region. Adding a second point, here with negative density, results in a nonuniform shape. Twenty points yields complex, smooth freeform shapes. This concept easily extends to three dimensions. (a) One Gaussian point. (b) Two Gaussian points. (c) Twenty Gaussian points.

II. Algorithm Details

Evolutionary algorithms are widely used across many disciplines for many different optimization problems. A good introduction to this class of algorithms may be found in [31]. Using the Gaussian mixtures representation, the genome is simply a list of points with six parameters each: the *X*, *Y*, and *Z* coordinates, density, falloff distance, and material index. Once the representation has been defined, there are still several choices to be determined regarding how the design automation algorithm can generate new designs from preexisting designs. The algorithm must have defined methods for crossover (combining two preexisting designs into a new design) and mutation (making small changes to an existing design).

For the Gaussian mixtures representation, a one-plane section crossover method was used. For each crossover event, a single plane is randomly generated that intersects the workspace. The new design consists of all the Gaussian points from one side of the plane from the first parent, as well as all the Gaussian points from the other side of the plane from the second parent. This method has the ability to preserve physical regions with helpful functionality. Mutating the Gaussian mixtures representation involves making small changes to any of the parameters of a few Gaussian points and occasionally adding or removing points. A mutation rate of 20% was used.

An evolutionary algorithm must also have some means of evaluating the goodness, or fitness, of each new design that it produces. This is very specific to the domain of the problem being solved. Since the algorithm that is presented here optimizes shapes for their properties in the physical world, a physical simulation must naturally be involved to guide the search process. For both the static cantilever beams and the soft robots, custom simulation software was developed to efficiently do both static and dynamic analysis. In both cases, the continuous domain that is generated by the Gaussian mixtures was discretized into voxels for the purpose of simulation. Voxels were chosen over a freeform mesh because they more naturally handle simulating multiple interspersed materials. As long as a sufficiently high resolution is chosen to capture the function of the object, this discretization does not adversely affect the evaluation or construction of continuous shapes.

A. Static Analysis

The cantilever beam designs were evaluated using the linear direct stiffness method. Materials are assumed to be well approximated by a linear isotropic material model and the geometry of the beam is assumed to not change significantly. To evaluate the fitness of a beam, the layer of voxels on the $-Y$ plane (base of the cantilever beam) were fixed to ground, while the layer of voxels at the other end of the beam had a downward force applied to them.

Each voxel has an associated stiffness and Poisson's ratio. Each voxel was modeled with all six translational and rotational DOFs. By calculating the effective translational and rotational stiffnesses of each link between voxels, a $6n \times 6n$ stiffness matrix $[K]$ can be assembled, where n is the total number of voxels in the simulation. This stiffness matrix relates the relative displacements D_n and angles θ_n of each voxel to the applied forces F_n and moments M_n. This forms a matrix equation

$$\begin{bmatrix} F_n \\ M_n \end{bmatrix} = [K] \begin{bmatrix} D_n \\ \theta_n \end{bmatrix}. \tag{1}$$

To obtain the resulting displacements and angles, this system of equations must be solved. Fortunately, the stiffness matrix is extremely sparse, which allows for structures with tens of thousands of DOFs to be solved in less than a second on an ordinary desktop computer. This sparse stiffness matrix problem was solved using the highly optimized PARDISO solver [32] to

yield the resulting displacements of each voxel of the structure under load.

B. Dynamic Analysis

The soft robots were evaluated using a more general dynamics simulator that is capable of modeling momentum effects and large deformations of soft materials [4]. Again, each voxel has a specific stiffness and Poisson's ratio associated with it. In addition, the density and coefficients of static and dynamic friction also were included for each material. The system was simulated using an iterative method. Forces between all voxels were calculated, then positions were updated synchronously using numerical integration. This allows for the inclusion of nonlinear effects, such as a soft shape bending over double or Coulomb stick–slip friction that are not possible to accommodate using the direct stiffness method. The friction between the soft robot and the floor was implemented in a manner that is consistent with the basic Coulomb friction model: any voxel touching the floor was either stopped or in a state of motion that is relative to the floor. Stopped voxels resisted any lateral motion until the lateral force exceeded the product of the normal force and the coefficient of static friction. Voxels in motion were slowed by a frictional force according to the coefficient of dynamic friction multiplied by the current normal force.

Individual interactions between voxels were modeled as beams according to the Euler–Bernoulli beam theory, with physical parameters set to the appropriate values based on the size of the voxels and the distance between them. Again, 6 DOF per voxel were used to enable both translational and rotational freedom. Volumetric actuation was achieved in simulation by simply changing the rest length between adjacent actuating voxels in all three dimensions. It was assumed that the volume changing actuation effects were fast compared with the response of the structure itself. Thus, the propagation and equalization of air pressure was not modeled. In addition, a collision detection system was in place to ensure that the soft robots did not self-penetrate.

C. Evolutionary Algorithm Parameters

The method of selecting which designs to combine and the criteria for when to keep a new design has a significant effect on the success of a evolutionary algorithm. Here, we use the deterministic crowding selection method [33]. In this method, two random individuals from the current population are selected for breeding. After the crossover process is completed to generate a candidate offspring design, the amount of evolutionary contribution from each parent is evaluated to determine the dominate parent. After a 20% chance of mutation for each parameter, the candidate solution is evaluated using the physics simulator. If the child design is more fit than its dominate parent, the child replaces this parent in the current population. Otherwise, the new design is discarded. This method allows for relatively small population sizes, while maintaining diversity. In Table II, we summarize the details of the evolutionary algorithm.

TABLE II
SUMMARY OF EVOLUTIONARY ALGORITHM PARAMETERS

Parameter	Value
Population Size	50
Selection method	Deterministic crowding
P(Mutation)	0.2
Solution Encoding	Gaussian Points

III. STATIC STRUCTURE RESULTS

Before applying these design automation techniques to soft robots, their capability to design heterogeneous multimaterial structures was verified by designing cantilever beams to deflect in to prescribed shapes under downward load. A slender cantilever beam with a homogeneous material distribution deflects in a polynomial cubic curve. By incorporating a finite thickness for the beam and a combination of hard and soft materials, it is possible to change this deflected profile into other smooth downward sloping profiles based on the material distribution (see Fig. 3).

In order to facilitate the emergence of the correct qualitative shapes, differences in quantitative deflection of competing designs were not penalized. In order for the design algorithm to evaluate each design, the beam was loaded and simulated to estimate the deflected shape. Both desired and simulated profiles were normalized so that the free end had unit deflection. Then, points along top surface of each deflected beam were recorded and compared with the desired profile. The workspace of the beam was selected to be $12 \times 12 \times 40$ (5760 total) voxels. With a domain this much large, the simulation is the computational bottleneck of the design automation algorithm. Because the Gaussian mixtures representation can be discretized at any arbitrary resolution, the design process was accelerated by first evaluating designs at low resolution ($6 \times 6 \times 20$ voxels). After the algorithm began to converge on good solutions, the algorithm automatically switched to the higher resolution evaluation. A single fitness metric was calculated for each beam regardless of resolution as the sum of the squared differences between each voxel on the top surface and the desired top-surface profile.

A total of four different beam profile shapes were tested in addition to verifying the deflected shape of the homogeneous beams. In approximate increasing order of difficulty, these include a straight profile, a circular-arc-shaped profile, a discontinuous-slope profile, and a fourth-order-polynomial profile that incorporated negative (upward) curvature. After running for several thousand generations, the design automation algorithm was able to match the simulated shape with a high degree of accuracy to each desired profile.

In order to verify the performance of the design automation in the physical world, each beam was autonomously fabricated in 3-D using an Objet Connex 500 printer. First, a homogeneous beam was printed and tested to verify the experimental procedure (see Fig. 4) with favorable results. Because the shapes were only being compared qualitatively, the absolute stiffness of the physical materials did not have to be exact. However, the ratio of stiffnesses of the two materials was important to replicate the designed shapes in the physical world. A combination of

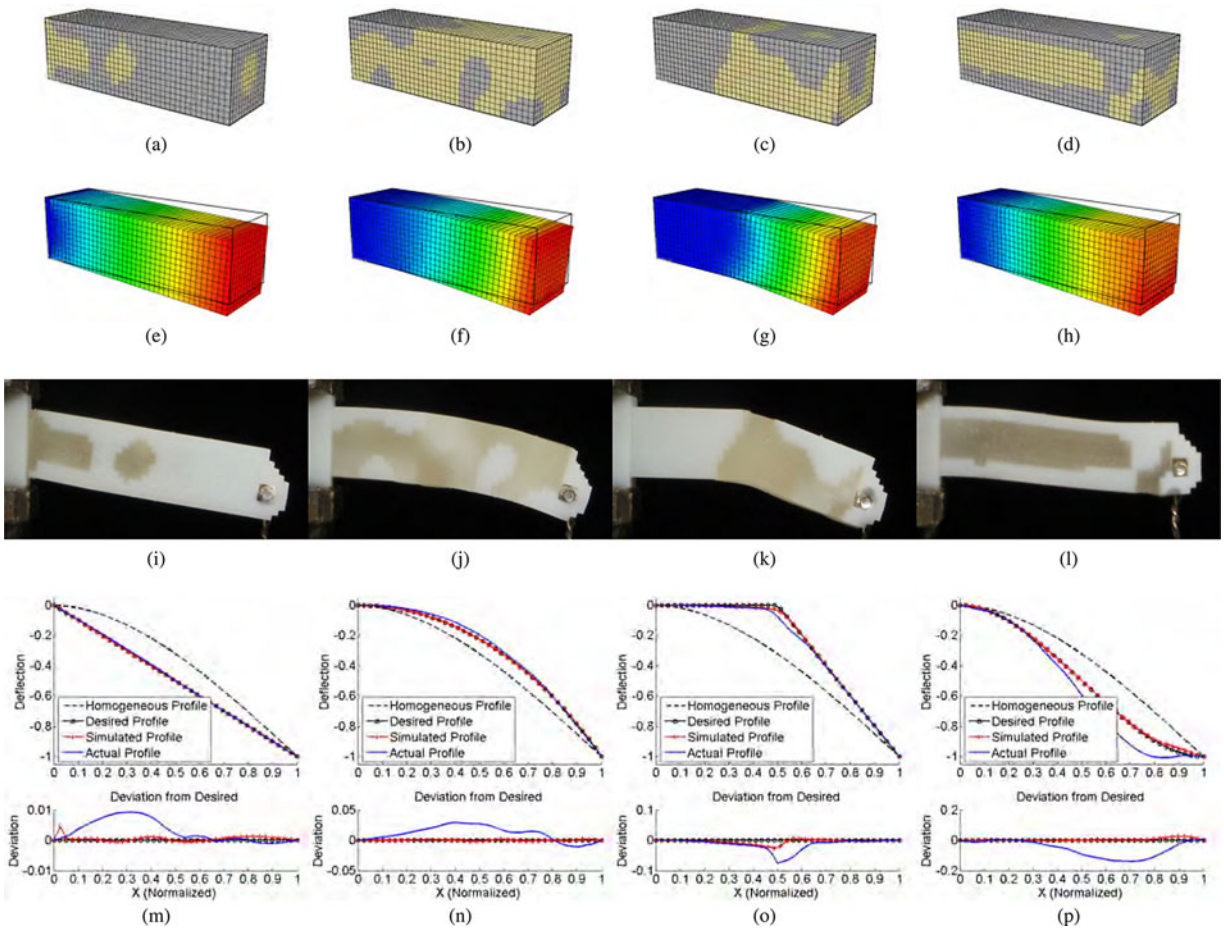

Fig. 3. Autonomously designed material distributions of static cantilever beams deflect in nontraditional top-surface profiles, such as a straight line (first column), circular arc (second column), discontinuous slope (third column), and fourth-order polynomial (fourth column). The generated material distributions (a)–(d) and the corresponding simulated deflected profiles (e)–(h) closely match the input functions. Actual beams were autonomously fabricated and tested (i)–(l). The results for both simulation and reality are compared with the desired shape and nominal shape (m)–(p).

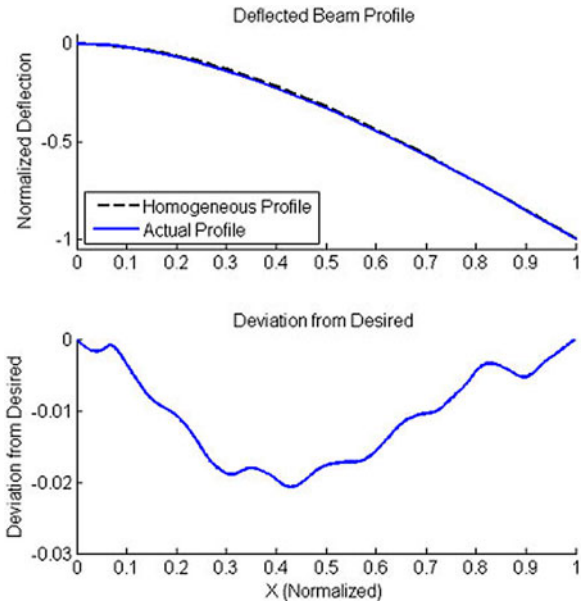

Fig. 4. Test beam of homogeneous material distribution was fabricated to verify the experimental setup. The rms error in deviation from the expected profile is about 1.25%.

Objet's VeroGrey (hard) and TangoPlus (soft) were used. Before printing, additional geometry was appended to each shape to facilitate physical testing. This included a base block to be clamped to ground and loops to enable force to be evenly applied at the free end.

Each design was printed out and tested. In order to compare the actual resulting physical beams with the expected results, we used image analysis to analyze the deflected profile of each beam. The test beam was firmly clamped at the fixed end, while enough downward force was manually applied to the free end to achieve a nontrivial deflection. An image of the beam was then captured from the side with the lighting and background setup to facilitate the accurate edge detection.

The process of extracting the profile of the beam from each image is shown in Fig. 5. First, an edge-finding filter was applied in Photoshop. The result was then thresholded into a black and white image. Extraneous image noise not connected to the profile of the beam was eliminated. A MATLAB script then determined the average height of the black pixels for each column of the image to create a 1-D function. This function was then smoothed using the locally weighted scatterplot smoothing (LOWESS) method with a linear fit over a span of 101 pixels

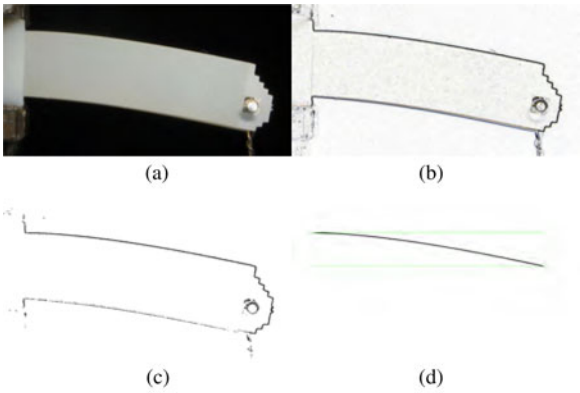

Fig. 5. To extract the profile of a deflected beam, an image of each beam was captured in its deflected state. Edge-finding filter was applied followed by a thresholding filter to create a black and white image. Pixels that are not associated with the top surface of the beam were then manually erased to allow the profile to be detected and segmented with a MATLAB script. (a) Original image. (b) Edge find. (c) Threshold. (d) Segment.

TABLE III
RMS ERROR IN DEVIATION OF SIMULATED AND ACTUAL BEAMS

Desired Shape	RMS Simulation Error	RMS Physical Error
Nominal Homogeneous	-	1.25%
Straight	0.10%	0.47%
Circular Arc	0.10%	1.64%
Discontinuous Slope	0.86%	2.36%
4th Order Polynomial	0.90%	7.62%

in the *X*-direction. This sufficiently smoothes out the effects of image noise, while retaining the actual shape of the beam. The resulting function is then normalized and compared with the simulated and desired function and the rms error between each is calculated. The results are given in Table III.

IV. DYNAMIC FREEFORM ROBOT RESULTS

The same design automation algorithm was then applied to create locomoting freeform soft robots by simply changing the criteria for how a candidate design was calculated. Using the dynamic simulator described earlier, 10 actuation cycles were imposed on each design. The fitness was taken to be the distance, in the positive *X*-direction, the center of mass moved. Because the 3-D shapes are initialized in free space, each shape was allowed to settle under gravity before the actuation commenced. The center of mass was then calculated and stored for comparison with the ending center of mass. This eliminated the propensity of the algorithm to "cheat" by creating designs that simply fell and flopped over in the right direction, but did not actually locomote.

In these experiments, the volumetrically actuated material varies sinusoidally in volume over time. The volume change was selected to be $\pm 20\%$. The relative speed (period) of this oscillation determines to what degree the dynamics of the structure play a role in the movement. Fast oscillations around the effective resonance of the shape will result in physical motion substantially out of phase with the input. Here, we selected the oscillation period to be significantly slower that any resonances so that the dynamic response of the structure does not dominate the motion. This is to more closely match the physical implementation that is used to demonstrate the freeform robot in the real world. We chose the representative values of 0.3 and 1 for dynamic and static friction, respectively.

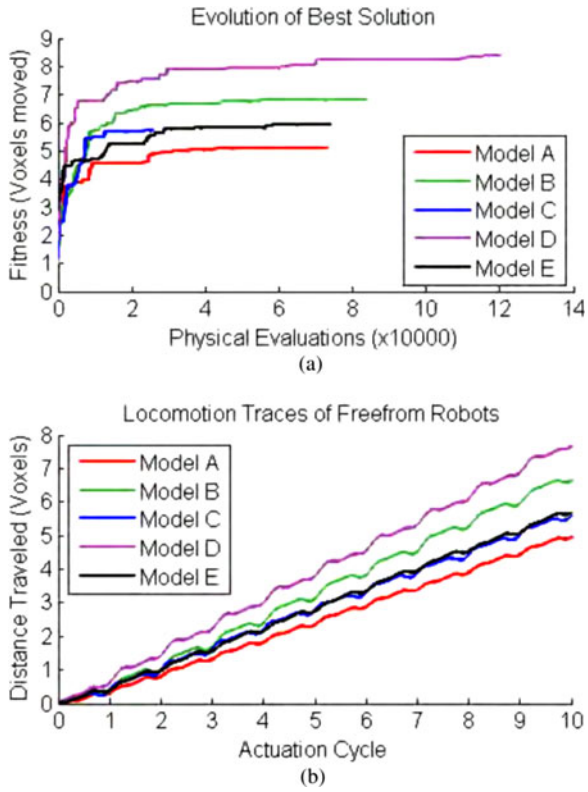

Fig. 6. Best results of five independent evolutionary algorithm runs designing freeform soft robots. (a) Fitness increases dramatically at first, then in smaller jumps corresponding to new discoveries, but eventually leveling off. (b) Movement of the best soft robot at the end of each run shows very similar locomotion pattern.

A resolution of $20 \times 20 \times 20$ (8000 possible) voxels was selected as a reasonable-sized domain to work within. The low-resolution bootstrapping of initial solutions was not utilized for the dynamic simulation because of abundant available computing power. Each random solution was initialized with 88 randomly generated Gaussian points, which was sufficient to create a range of interesting topologically varied shapes. Although mutation operators occasionally add or subtract points from this total, the final solutions generally had around 100 total points. The evolution history of five separate runs of the evolutionary algorithm is shown in Fig. 6(a). The solutions were run for an arbitrary amount of time on a single desktop computer, ranging from overnight to a couple of days, corresponding to tens of thousands of individual physics simulations.

All five solutions are shown in Fig. 7. The fundamental mode of locomotion is similar for all of them, as evidenced by very similar traces of position of the center of mass over time [see Fig. 6(b)]. All examples exhibit a scooting motion, where the expanding phase of the actuator pushes the soft robot forward, while the contraction phase resets the back of the robot to repeat. The distribution of the mass of the robot alternately weights and unweights the front and back of the robot as the material actuates, allowing it to make and break static friction at the

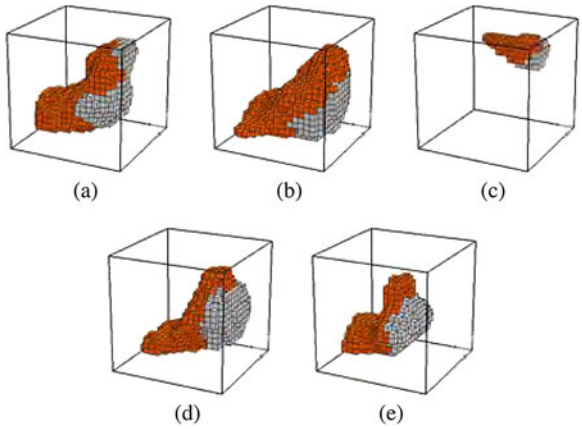

Fig. 7. Resulting freeform robots of the five evolution runs all converged to similar morphologies and locomotion modes for the physical parameters that are assigned in these experiments.

appropriate places and times to achieve locomotion. The similarity of the solutions demonstrates that for the given algorithm and physical parameters that are used, there is a clear evolutionary path from the more complex and topologically varied random initializations to the much simpler class of optimized designs that resulted. Although these cannot be guaranteed to be a global optimum, the convergent morphologies demonstrate that these solutions are at least a very broad local optimum for the physical parameters that are used in these tests.

Different physical parameters would likely yield other morphologies and modes of locomotion. The parameters that are used here are analogous to a very soft gelatinous substance. This favors the scooting motion observed because there is a high ratio of floor-contact surface area to volume, tending to discourage legged creatures, which would have a much lower ratio. Utilizing stiffer lighter materials would introduce the possibility of legged locomotion, but further work is needed to explore the different classes of solutions that lie within this broad parameter space.

Of the five designs, model B was selected for fabrication. A significant missing link in demonstrating true freeform soft robots is the current inability to 3-D print volumetric actuators. In order to physically demonstrate the design that is presented here, we rely on modulating the external environment in order to achieve selective volumetric expansion and contraction of autonomously fabricated regions. By varying the pneumatic pressure of the environment, regions of the soft robot that are pneumatically isolated from the environment experience a change in relative pressure. Because the surrounding material is soft, this results in a change of volume. Regions of the robot that are pneumatically connected to the environment can quickly equalize to the changing external pressure; therefore, no volume change is observed.

Here, we used a custom-layered manufacturing process using preexisting open-cell and closed-cell foam rubbers to fabricate the robots. A routine was written to send the profile of each voxel layer to a laser cutter to be autonomously cut out, along with alignment holes between layers. The individual layers were then

TABLE IV
COMPARISON OF SIMULATED AND PHYSICAL SOFT ROBOTS

	Simulated	Physical
Size (mm)	18	55*
Weight (grams)	1.675×10^{-3}	12.18*
Actuation Period (s)	0.07	10
Effective Dynamic Actuation Force (μN)	**6.15**	**6.09**
Speed (% Body length/actuation cycle)	3.38%	3.96%

*The size and weight of the physical robot were determined by the availability of suitable construction materials. Actuation period was adjusted accordingly to yield a similar effective dynamic actuation force.

assembled onto alignment pins with glue interspersed between layers to finish the 3-D structure. Although this process is not yet fully autonomous, human involvement is minimal, and no particular skill is needed to stack and assemble the fabricated layers.

The robot was chosen to be 55 mm in length in order to use commonly available foam rubber sheet stock. Based on the the simulated size, density, and actuation period of the evolved solutions, this size gave a similar scale of relative dynamic internal forces when actuated with a period of approximately 10 s. The effective dynamic actuation force F_d for each case was calculated according to

$$F_d = m|a| \tag{2}$$

where m is the mass of the total robot, and $|a|$ is the magnitude of the acceleration assuming sinusoidal actuation. Given an actuator displacement d that is proportional to the size of the robot and an actuation frequency f, an effective position P can be modeled as

$$P(t) = d\sin(ft) \tag{3}$$

which when differentiated twice yields

$$A(t) = df^2\sin(ft) \tag{4}$$

where A is the effective acceleration. Thus, the magnitude of acceleration $|A|$ is

$$|A| = df^2 \tag{5}$$

so that

$$F_d = mdf^2. \tag{6}$$

This effective dynamic actuation force is not quantitatively meaningful, but for a given geometry of soft robot, it provides a qualitative measure of the similarity of behavior under sinusoidal actuation. For the parameters used here, the simulation soft robot had an F_d value of 6.15 μN, and the actual robot had a value of 6.09 μN (see Table IV). An acrylic floor was chosen for the physical experiments. When coupled with the silicone foam rubber of the soft robot, the interface exhibited the expected stick–slip frictional characteristic utilized in the simulation. The exact tuning of frictional coefficients was determined to not be critical to the performance of the robot since the locomotion pattern depended primarily on making and breaking static friction.

The physical robot was placed in a pressure and vacuum chamber to verify the locomotion (see Fig. 8). The trace of the position of the physical robot at the end of each actuation cycle

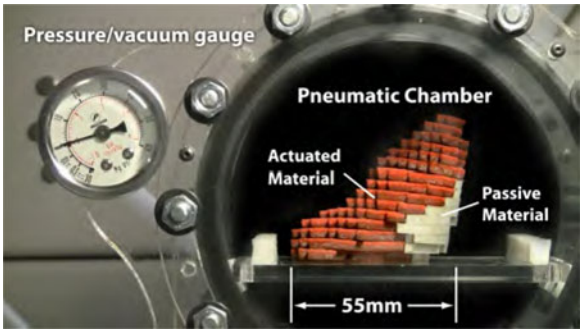

Fig. 8. Soft robot was placed in a pneumatic pressure/vacuum chamber to demonstrate locomotion. Selective volumetric actuation was attained using closed cell (orange) foam rubber, which changes volume as the air pressure changes periodically. Open cell foam rubber (white) does not change volume.

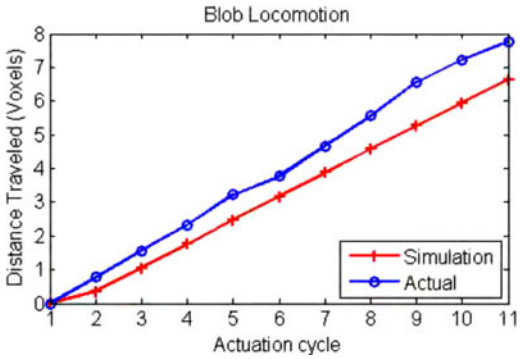

Fig. 9. Movement of the simulated and actual model B soft robot shows similar velocities, suggesting that the physics simulator is an accurate predictor of real-world performance.

is plotted with the simulation results (see Fig. 9). In each case, the distance traveled is normalized by the size of the voxels. The physical robot displays robust locomotion and progresses at a very similar rate as the simulation predicts.

The ability of the physical robot to match the simulation velocity can be attributed to several key parameters. The most important by far is the percent volumetric actuation. If the actuation is not large enough to ever break the static friction with the floor, the robot will never move. Once the actuation amount has reach this threshold, the velocity of the robot per actuation cycle will increase as the amount of actuation increases. By extension, as long as the available actuation force is large compared with the frictional forces and the friction coefficients are such that static friction is in effect at the appropriate periods of each cycle, the exact friction coefficient values are not critical. The combined passive and actuator material properties dictate the change in pressure that is necessary to achieve a given actuation amount, but once this is calibrated to the simulation, the stiffness of these materials plays a secondary role in semistatic locomotion that was specified by the optimization algorithm. A composite image of three frames of a video shows the physical robot locomoting under the varying relative pressure (see Fig. 1).

V. Conclusion

For decades of mechanisms and robotics, the paradigm of discrete rigid links and single-DOF joints has prevailed. Here, the effectiveness of evolutionary algorithms paired with the Gaussian mixtures representation to autonomously design freeform shapes and material distributions to achieve high-level functionality in the physical world has been demonstrated. The same algorithm can be applied to very different mechanical design problems by simply changing the function that evaluates: How good a potential design is? This is easily extended to exploring multiple, competing objectives by incorporating a Pareto front of solutions.

In the future, we envision directly manufacturing pneumatically actuated soft robots with additive manufacturing techniques. By utilizing soft, elastic material, discrete hollow cells could be fabricated. Those which are sealed off from the exterior environment would act as actuators and others which are open to the environment would be passive. By the incorporation of a source of pressure connected to the sealed actuation regions, the designs that are presented here would work without adaptation in our everyday environment, where atmospheric pressure is constant. This would also allow more complex actuation patterns, such as out-of-phase actuation of different regions of the soft robot. With such distributed actuation, morphological computation could also be explored for the purpose of controlling gaits.

With the unconstrained design space enabled by multimaterial additive manufacturing techniques, a whole new design space of soft, amorphous robots is possible. There are still challenges that remain regarding how to effectively design functional objects with discrete components and how to best actuate a truly amorphous soft robot. We believe that volumetric actuation provides a viable method for actuating soft robots and offers many advantages for design automation because locomotion is dictated purely by material distributions. However, more work is needed to develop self-contained and powered volumetrically actuating materials. As these challenges are addressed, robots will be better equipped to operate robustly in highly uncertain and dangerous environments.

References

[1] M. P. Bendsoe and N. Kikuchi, "Generating optimal topologies in structural design using a homogenization method," *Comput. Methods Appl. Mech. Eng.*, vol. 71, no. 2, pp. 197–224, 1988.

[2] J. Hiller and H. Lipson, "Multi-material topological optimization of structures and mechanism," in *Proc. Genetic Evol. Comput. Conf.*, 2009, pp. 1521–1528.

[3] J. Hiller and H. Lipson, "Evolving amorphous robots," in *Proc. 12th Int. Conf. Artif. Life*, 2010, pp. 717–724.

[4] J. Hiller and H. Lipson, "Dynamic simulation of soft heterogeneous objects," 2012, to be published.

[5] J. Rieffel, B. Trimmer, and H. Lipson, "Mechanism as mind: What tensegrities and caterpillars can teach us about soft robotics," in *Proc. 11th Int. Conf. Artif. Life*, 2008, pp. 506–512.

[6] A. Mozeika, E. Steltz, and H. M. Jaeger, "The first steps of a robot based on jamming skin enabled locomotion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Oct. 11–15, 2009, pp. 408–409.

[7] Y. Bar-Cohen, *Electroactive Polymer (EAP) Actuators as Artificial Muscles—Reality, Potential and Challenges*, vol. PM98. Bellingham WA: SPIE, 2001.

[8] J. Rieffel, F. Saunders, S. Nadimpalli, H. Zhou, S. Hassoun, J. Rife, and B. Trimmer, "Evolving soft robotic locomotion in physx," in *Proc. Genetic Evolutionary Comput. Conf.*, New York: ACM, 2009.
[9] Y. Sugiyama and S. Hirai, "Crawling and jumping by a deformable robot," *Int. J. Robot. Res.*, vol. 25, no. 5–6, pp. 603–620, 2006.
[10] B. A. Trimmer, A. E. Takesian, B. M. Sweet, C. B. Rogers, D. C. Hake, and D. J. Rogers, "Caterpillar locomotion: A new model for softbodied climbing and burrowing robots," in *Proc. 7th Int. Symp. Technol. Mine Problem*, Monterey, CA, 2006.
[11] F. Ilievski, A. D. Mazzeo, R. F. Shepherd, X. Chen, and G. M. Whitesides, "Soft robotics for chemists," *Angewandte Chemie Int Ed.*, vol. 50, pp. 1890–1895, 2011.
[12] J. J. Beaman, H. L. Marcus, D. L. Bourell, J. W. Barlow, R. H. Crawford, and K. P. McAlea, *Solid Freeform Fabrication: A New Direction in Manufacturing*, Norwell, MA: Kluwer, 1997.
[13] E. Malone, K. Rasa, D. Cohen, T. Isaacson, H. Lashley, and H. Lipson, "Freeform fabrication of zinc-air batteries and electromechanical assemblies," *Rapid Prototyp. J.*, vol. 10, no. 1, pp. 58–69, 2004.
[14] E. Malone and H. Lipson, "Fab@home: The personal desktop fabricator kit," *Rapid Prototyp. J.*, vol. 13, no. 4, pp. 245–255, 2007.
[15] Objet. (2010). Objet geometries Inc. [Online]. Available: http://www.objet.com
[16] H. Lipson and J. B. Pollack, "Automatic design and manufacture of robotic lifeforms," *Nature*, vol. 406, no. 6799, pp. 974–978, 2000.
[17] J. B. Pollack, H. Lipson, G. Hornby, and P. Funes, "Three generations of automatically designed robots," *Artif. Life*, vol. 7, no. 3, pp. 215–223, 2001.
[18] A. Diaz and R. Lipton, "Optimal material layout for 3D elastic structures," *Struct. Multidiscipl. Optim.*, vol. 13, no. 1, pp. 60–64, 1997.
[19] P. Fernandes, J. M. Guedes, and H. Rodrigues, "Topology optimization of three-dimensional linear elastic structures with a constraint on perimeter," *Comput. Struct.*, vol. 73, no. 6, pp. 583–594, 1999.
[20] M. J. Buehler, B. Bettig, and G. G. Parker, "Topology optimization of smart structures using a homogenization approach," *J. Intell. Mater. Syst. Struct.*, vol. 15, no. 8, pp. 655–667, 2004.
[21] S. Nishiwaki, M. I. Frecker, S. Min, and N. Kikuchi, "Topology optimization of compliant mechanisms using the homogenization method," *Int. J. Numer. Methods Eng.*, vol. 42, no. 3, pp. 535–559, 1998.
[22] O. Sigmund and S. Torquato, "Design of smart composite materials using topology optimization," *Smart Mater. Struct.*, vol. 8, pp. 365–379, 1999.
[23] M. J. Jakiela, C. Chapman, J. Duda, A. Adewuya, and K. Saitou, "Continuum structural topology design with genetic algorithms," *Comput. Methods Appl. Mech. Eng.*, vol. 186, no. 2–4, pp. 339–356, 2000.
[24] C. Kane, "Topological optimum design using genetic algorithms," *Control Cybern.*, vol. 25, no. 5, pp. 1059–1088, 1996.
[25] C. Kane and M. Schoenauer, "Genetic operators for two-dimensional shape optimization," in *Proc. Artif. Evol.*, 1996, pp. 355–369.
[26] R. Kicinger, T. Arciszewski, and K. D. Jong, "Evolutionary computation and structural design: A survey of the state-of-the-art," *Comput. Struct.*, vol. 83, no. 23–24, pp. 1943–1978, 2005.
[27] K. Sims, "Evolving virtual creatures," in *Proc. SIGGRAPH*, New York, ACM, Jul. 24–29, 1994, pp. 15–22.
[28] J. A. Sethian and A. Wiegmann, "Structural boundary design via level set and immersed interface methods," *J. Comput. Phys.*, vol. 163, pp. 489–528, 2000.
[29] M. Y. Wang, X. Wang, and D. Guo, "A level set method for structural topology optimization," *Comput. Methods Appl. Mech. Eng.*, vol. 192, no. 1–2, pp. 227–246, 2003.
[30] F. Pernkopf and D. Bouchaffra, "Genetic-based EM algorithm for learning Gaussian mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1344–1348, Aug. 2005.
[31] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. New York: Addison-Wesley, 1989.
[32] O. Schenk and K. Gartner, "Solving unsymmetric sparse systems of linear equations with pardiso," *Future Gener. Comput. Syst.*, vol. 20, no. 3, pp. 475–487, 2004.
[33] S. Mahfoud, "Niching methods for genetic algorithms," Ph.D. dissertation, Dept. Comput. Sci., Univ. Illinois, Urbana, IL, 1995.

Jonathan Hiller received the B.S. degree in mechanical engineering from the University of Washington, Seattle, in 2006 and the M.S. and Ph.D. degrees in mechanical engineering in 2009 and 2011, respectively, from Cornell University, Ithaca, NY.

He was a Member of the Cornell University Computational Synthesis Laboratory. He was a member of the Sibley School of Mechanical and Aerospace Engineering, Cornell University. His research interests include the boundary between the physical and digital world, specifically, the automated design and fabrication of voxel-based materials and functional systems.

Dr. Hiller received the National Science Foundation Graduate Research fellowship.

Hod Lipson (M'02) received the B.Sc. degree in mechanical engineering and the Ph.D. degree in mechanical engineering in computer-aided design and artificial intelligence in design from the Technion—Israel Institute of Technology, Haifa, Israel, in 1989 and 1998, respectively.

He is currently an Associate Professor with the Mechanical and Aerospace Engineering and Computing and Information Science Schools, Cornell University, Ithaca, NY. He was a Postdoctoral Researcher with the Department of Computer Science, Brandeis University, Waltham, MA. He was a Lecturer with the Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, where he was engaged in conducting research in design automation. His current research interests include computational methods to synthesize complex systems out of elementary building blocks and the application of such methods to design automation and their implication toward understanding the evolution of complexity in nature and in engineering.