EvoFab: A Fully Embodied Evolutionary Fabricator

John Rieffel and Dave Sayles

Union College Computer Science Department
Schenectady, NY 12308 USA

Abstract. Few evolved designs are subsequently manufactured into physical objects – the vast majority remain on the virtual drawing board. We suggest two sources of this "Fabrication Gap". First, by being *descriptive* rather than *prescriptive*, evolutionary design runs the risk of evolving interesting yet *unbuildable* objects. Secondly, in a wide range of interesting and high-complexity design domains, such as dynamic and highly flexible objects, the gap between simulation and reality is too large to guarantee consilience between design and object. We suggest that one compelling alternative to evolutionary design in these complex domains is to avoid both simulation and description, and instead evolve artifacts directly in the real world. In this paper we introduce EvoFab: a fully embodied evolutionary fabricator, capable of producing novel objects (rather than virtual designs) *in situ*. EvoFab thereby opens the door to a wide range of incredibly exciting evolutionary design domains.

1 Introduction

Evolutionary algorithms have been used to design a wide number of virtual objects, ranging from virtual creatures [12] to telescope lenses [1]. Recently, with the advent of rapid prototpying 3-D printers, an increasing number of evolved designs have been fabrictated in the real world as well. One of the earliest examples of an evolved design crossing the "Fabrication Gap" into reality is Funes' LEGO structures [4]. In this work, the genotypes were a direct encoding of the physical locations of bricks in the structure - a virtual "blueprint" of the design. Fitness, based upon the weight-bearing ability of the structures, was determined inside a quasi-static simulator. The authors were able to translate the virtual phenotype into a physical object by reading the blueprint and placing physical bricks accordingly.

Another notable example of a manufactured evolved design is Lohn's satellite antenna [6]. The genotype in this case was a generative encoding L-system which, when interpreted by a LOGO-like "turtle", drew a 3-D model of the antenna. Fitness was determined by measuring the performance of the design within an off-the-shelf antenna simulator.

Other evolved designs to cross the Fabrication Gap include robots [7], furniture [5], and tensegrity structures [10]. In each of these later cases, phenotypes were 3D CAD models which could then be printed directly by rapid protyping 3D printers.

The quality of these examples belies their quantity. The vast majority of evolved designs remain on the virtual “drawing board”, never to be manufactured. A closer analysis of the examples above provides some insight into this “Fabrication Gap”. For Funes work, building a physical LEGO structure from a descriptive blueprint was facilitated, at least in principle, by the close correspondance between virtual and physical LEGO bricks. In practice, however, the blueprints alone didn’t contain sufficient assembly information: particularly for large structures, the evolved designs first had to be assembled on a flat horizontal surface and then tilted into place – an operation that cannot be inferred from a blueprint. In Lohn’s antenna work, the final product was manufactured by hand: using the 3D model as a guide, a skilled antenna engineer manually bent and soldered pieces of conductive wire to the specified lengths and angles. As these 3D antenna models become more complex, this process becomes increasingly intractable.

We see two primary sources of this “Fabrication Gap” between evolved virtual design and physical object. The first issue is that, conventionally, evolved designs are purely *descriptive*. By specifying what to build but not *how to build it*, evolutionary design runs the risk of evolving interesting yet *unbuildable* objects. Imagine an evolutionary design system which evolves images of chocolate cakes. The image describes what the final product looks like (which may be delicious), but there is nothing in the image which provides insight into *how* it should be prepared, or whether it can even be prepared at all. Similarly, a descriptive representation shows a finished product, but contains no information about how to manufacture it.

Secondly, the evolutionary design of complex objects requires high fidelity simulation in order to guarantee that the physical manifestation behaves like its virtual counterpart. For static and rigid objects, such as the tables and robot parts mentioned above, fabrication is relatively straight forward: their behavior can be realistically simulated, and their descriptive phenotype is easily translated into a print-ready CAD file. However, for high-complexity design domains, such as dynamic and highly flexible objects, the gap between simulation and reality is too large to reliably manufacture designs evolved in simulation.

This begs the question: in these high complexity domains, is it at all possible to dispense with simulation and description entirely, and instead evolve assembly instructions directly within a rapid prototyper? In such an “evolutionary fabrication” scenario the genotype consists of a linear encoding of instructions to the printer, and the evaluated phenotype is the resulting structure. These ideas have been motivated and explored using *simulations* of rapid prototypers [8] [9], but until now haven’t been instantiated in the real world.

On the face of it of course this proposition seems extreme, and the reasons against it are obvious. First of all, rapid prototyping is a slow process, and so an evolutionary run of hundreds (even thousands) of individuals might take days or weeks – not to mention the associated cost in print material. Furthermore, commercial rapid prototypers cost hundreds of thousands of dollars, and do not allow the access to their underlying API which this approach requires. Finally, commercial prototypers typically only print relatively rigid materials, and so are incapable of producing objects from more interesting design domains.

Fortunately, the recent advent of inexpesive desktop fabricators allows for a re-examination of these constraints. Hobbyist-oriented units, such as the Fab@Home and the Makerbot Cupcake, cost only a couple thousand dollars assembled, are open source and "hackable" and, most importantly, are capable of printing a much wider range of print media - from wax and silocone elastomers to chocolate.

Furthermore, evolutionary embedded in the real world has produced some profoundly interesting results in other domains. Consider for instance Thompson's seminal work on "Silicon Evolution" [13], in which pattern discriminators evolved directly on an FPGA behaved qualitatevely differently than those evolved in simulation. In fact, the final product wound up exploiting thermal and analog properties of the FPGA – something well outside the domain of the simulator. Similarly, Watson and Ficici applied "Embodied Evolution" [15] (their term) to a population of simple robots, and produced neural network based control strategies which varied significantly from their simulated-evolution counterparts. In each case, the lesson has been that evolution directly in the real world can produce profound results which would have been *impossible* to produce via simulation. We draw our inspiration for Evolutionary Fabrication largely from these groundbreaking insights.

In this paper we introduce EvoFab: a fully embodied evolutionary fabricator, capable of automatically designing *and manufacturing* soft and dynamic structures, thereby bridging the "Fabrication Gap". After describing the design of this unit in detail, we demonstrate proof-of-concept Evolutionary Fabrication of flexible silicone objects. The ability to automatically design and build soft and dynamic structures via EvoFab opens the door to a wide range of exciting and vital design domains, such as soft robots and biomedical devices.

2 EvoFab: An Evolutionary Fabricator

The system capable of embodied evolutionary fabrication (EvoFab) consists of two parts: a Fab@Home desktop rapid prototyper [14], and a python-based genetic algorithm which interfaces with the Fab@Home. The Fab@Home printer (Figure 1) was developed as a hobbyist desktop rapid prototyper. Its low price, open source software, and large range of print materials makes it ideally suited as an Evofabber.

A print syringe, mounted on a X-Y plotter, extrudes material onto an 8" square Z-axis-mounted print platform. We specify seven specific operations which the printer may perform:

in,out - move the print head in the $+/-$Y direction 3mm
left,right - move the print head in the $+/-$X direction 3mm

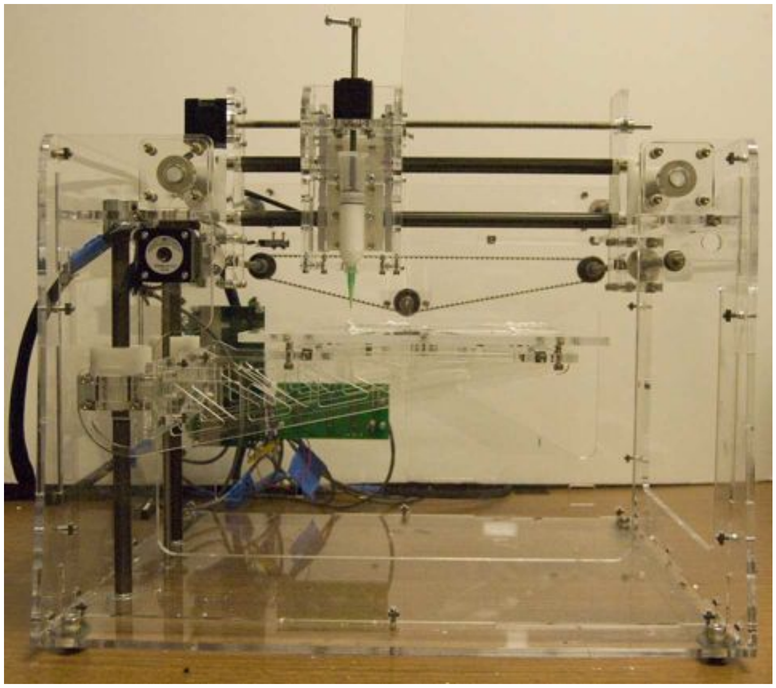

Fig. 1. A Fab@Home desktop prototyper is used as the foundation of the EvoFab

up,down - move the print head in the $+/-Z$ direction 3mm
extrude - pushes a fixed volume of print media through the 0.8mm syringe.

We refer to a linear encoding of these operations as an *assembly plan*.

In conventional setups, prototypers produce three dimensional objects by printing successive layered "slices" of the object on the horizontal plane, lowering the print platform between slices. In the context of Evolutionary Fabrication however, we prefer a more open-ended freeform approach, and so place no constraints upon the print process. The print head is free to move in almost any direction and to perform any operation during the execution of an assembly plan – even if that means causing the syringe to collide with the object it is printing. We will discuss why this might be beneficial in the last section of this paper.

After testing a variety of materials ranging from Play-Doh to alginate (an algae-based plaster), we selected silicone bath caulk (GE Silicone II) because of its relatively short (30-minute) cure time, and viscosity (it is thick enough to remain inside the print syringe until extruded, but thin enough to easily extrude into a single strand). Figure 2 illustrates the extrusion of silicone onto the print surface.

Because extruding material from a syringe creates a "thread" which dangles between the syringe and the print platform, an *extrude* command followed by a directional command such as $left$ will print a 3mm long line on the print

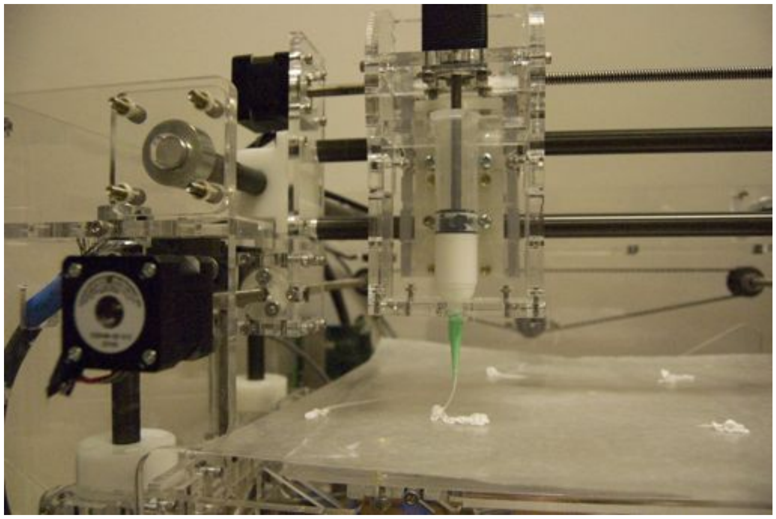

Fig. 2. Freeform three dimensional printing of silicone is accomplished by a syringe mounted to an X-Y plotter. The print platform can be moved vertically along the Z axis.

platform. An example assembly plan capable of printing a 3mm square of silicone might appear as the following:

$[extrude, left, extrude, in, extrude, right, extrude, out]$

In the context of evolutionary fabrication, these linear encodings of instructions form can a *genotype*. Mutation and crossover of genotypes is accomplished in just as it would be in any other linear encoding. Figure 3 illustrates the results of two assembly plan genotypes which differ by a small mutation.

It is worth emphasizing that assembly plans are an *indirect* encoding – the object which results from executing a particular assembly plan can be considered its *phenotype*. This layer of indirection gives rise to some interesting consequences, most significant is that there is no longer a $1 : 1$ mapping from genotype to phenotype (as there would be in a direct encoding, just as simple bit string GA). Rather, there is an $N : 1$ mapping: physically identical phenotypes can arise from distinct underlying genotypes. In fact, when you take into account the stochastic nature of the fabrication process it becomes an $N : N$ mapping, meaning that a single genotype can produce slightly different phenotypes when executed multiple times. We explore the consequences of this in our discussion below.

3 Proof of Concept: Interactive Evolution of Shape

We can demonstatrate the potential of evolutionary fabrication using a relatively simple Interactive Genetic Algorithm (IGA). Based upon Dawkin's idea of the

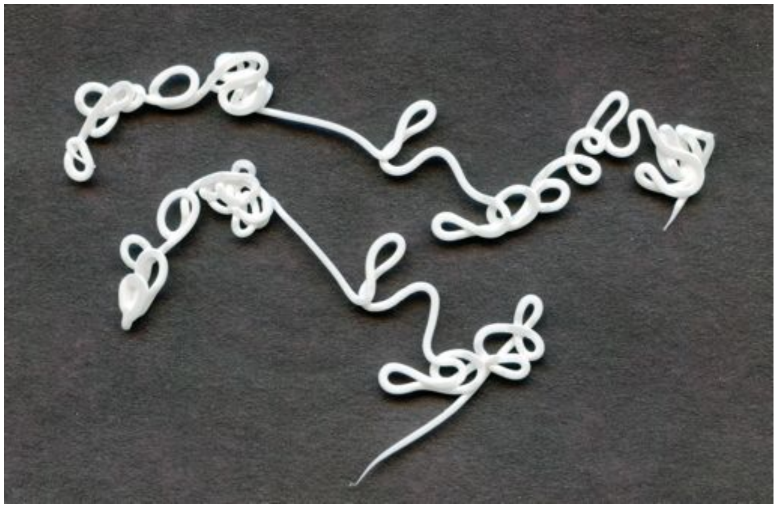

Fig. 3. Small changes to assembly plan genotypes produce corresponding changes to the resulting silicone phenotype. The image above compares an original (top) with its mutant (bottom) in which a trainling sequence of the assembly plan has been replaced. Each object was printed from right to left.

"Blind Watchmaker" [3], IGAs replace an objective and automated fitness function with human-based evaluaton. IGAs have been successful in a wide range of evolutionary design tasks, most notably in Karl Sims' seminal work on artificial creatures [12], [11].

We chose as a design task the simple evolution of circular 2-dimensional shapes. A population of size 20 was initialized with random assembly plans, each of which was 20 instructions long. Individuals were then printed onto the platter in batches of four. Once the population was completely printed, the 10 best (most circular) individuals were then selected as parents for the subsequent generations. New children were created using cut-and-splice crossover [16]

Each platter of four individuals took roughly 10 minutes to print, corresponding to slightly less than an hour of print time per generation.

Figure 4 compares sample phenotypes from the first and ninth generations. After only a small number of generations, the population is already beginning to converge onto more circular shapes.

4 Discussion

The results presented in our proof-of-concept evolutionary fabrication above are enough to lend credence to the potential of EvoFab for exploring even more interesting and complex design domains. Before discussing these applications in more detail it is first worth discussion some of the implications and limitations of this approach.

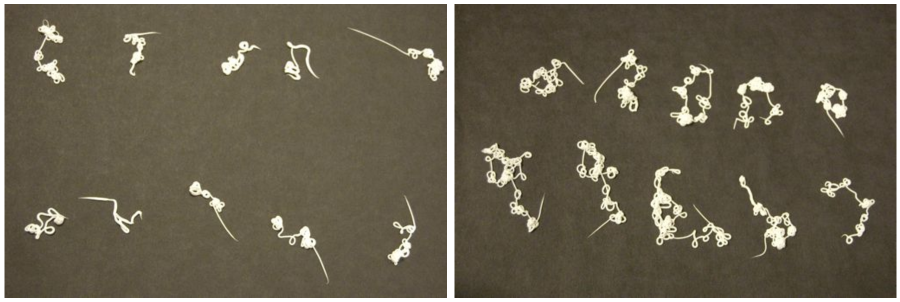

Fig. 4. Sample individuals from the first (left) and ninth (right) generationss of the interactive evolution in which the user is selecting for roundness of shapes. After relatively few generations the population is beginning to converge onto more circular shapes.

4.1 Fabrication and Epigenetic Traits

One of the more fascinating consequences of embodied evolutionary fabrication is the capacity for the system as a whole to produce *epigenetic traits* - that is, phenotypic characteristics which arise purely from the mechanics of assembly, and have no underlying genotypic source. Consider for example the phenotypes in Figure 5, in which the user was selecting for shapes resembling the letter 'A'. At a glance one would assume that the "cross" of the A shapes was produced by an explicit set of operations within the underlying genotypes. In fact, they are instead caused by the print head "dragging" an extraneous thread of print material across the genotype as it moves between print regions.

Explorations into simulated evolutionary fabrication have suggested that there may be some interesting benefits to this kind of phenomenon [8]. Consider for instance a print process which extruded two separate subassemblies and then used the syringe head to dynamically asseble them into a larger structure. We hope to use EvoFab to further explore the consquences in embodied systems as well.

4.2 Material Use and Conservation

A natural consequence of evolutionary fabrication is that a significant amount of print material is consumed over the multiple generations of phenotype evaluations. And, while silicone elastomer is less expensive than the plastics used in high-end commercial rapid prototypers, the costs still add up.

In order to address this issue we are exploring a number of alternative and recyclable materials such as wax and even ice [2]. Ideally, once they are evaluated for fitness, phenotypes could then be reduced to their original material for reuse in a subsequent print cycle.

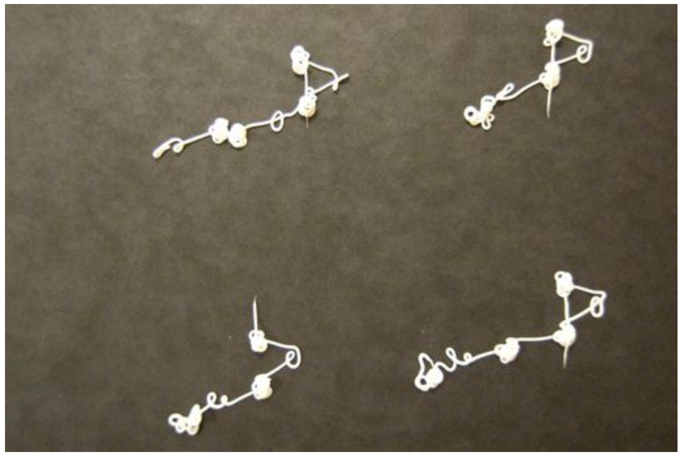

Fig. 5. Example of epigenetic traits in a set of phenotypes evolved for likeness to the letter 'A'. In each case, the "crosspiece" which connects the shorter leg to the longer leg is not caused by a genotypic sequence, but is instead caused by the print head dragging extra material across the phenotype as it finishes one print job and moves to the adjacent print region.

4.3 Design Domains

The domains in which Evolutionary Fabrication holds the most promise are those which are too complex or too inscrutable to realistically simulate. One such area is the design of flexible and dynamical systems, such as the morphology of completely soft robots.

In light of recent natural disasters in Haiti and Chile, there is a compelling need for more versatile and robust search and rescue robots. Imagine, for instance, a machine that can squeeze through holes, climb up walls, and flow around obstacles. Though it may sound like the domain of science fiction, modern advances in materials such as polymers and nanocomposites such a "soft robot" is becoming an increasing possibility.

Unfortunately, soft and deformable bodies can posess near-infinite degrees of freedom, and elastic pre-stresses mean that any local perturbation causes a redistribution of forces throughout the structure. As a consequence, soft structures are incredibly difficult to realistically simulate, even in non-dynamic regimes. Furthermore, there are no established principles or purely analytical approaches to the problem of soft mechanical design and control – instead the design task involves significant amounts of human-based trial and error.

EvoFab allows the power of evolutionary design techniques to be applied to this compelling and vital design domain. Soft bodies could be evolved and evaluated *in situ*, without resorting to simulation or post-hoc methods. The results of such endeavors could have significant consequences not just for search-and-rescue, but also in biomedical applications such as endoscopy.

References

1. Sameer H. Al-Sakran, John R. Koza, and Lee W. Jones. Automated re-invention of a previously patented optical lens system using genetic programming. In Maarten Keijzer, Andrea Tettamanzi, Pierre Collet, Jano I. van Hemert, and Marco Tomassini, editors, *Proceedings of the 8th European Conference on Genetic Programming*, volume 3447 of *Lecture Notes in Computer Science*, pages 25–37, Lausanne, Switzerland, 2005. Springer.
2. E. Barnett, J. Angeles, D. Pasin.i, and P. Sijpkes. Robot-assisted rapid prototyping for ice structures. In *IEEE Int. Conf. on Robotics and Automation*, 2009.
3. Richard Dawkins. *The Blind Watchmaker.* W. W. Norton & Company, Inc., September 1986.
4. Pablo Funes and Jordan B. Pollack. Evolutionary body building: Adaptive physical designs for robots. *Artificial Life*, 4(4):337–357, 1998.
5. Gregory S. Hornby and Jordan B. Pollack. The advantages of generative grammatical encodings for physical design. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC2001*, pages 600–607, COEX, World Trade Center, 159 Samseong-dong, Gangnam-gu, Seoul, Korea, 27-30 2001. IEEE Press.
6. J. D. Lohn, G. S. Hornby, and D. S. Linden. An Evolved Antenna for Deployment on NASA's Space Technology 5 Mission. In U.-M. O'Reilly, R. L. Riolo, T. Yu, and B. Worzel, editors, *Genetic Programming Theory and Practice II.* Kluwer, 2005.
7. Jordan B. Pollack, Hod Lipson, Gregory Hornby, and Pablo Funes. Three generations of automatically designed robots. *Artifial Life*, 7(3):215–223, 2001.
8. John Rieffel. *Evolutionary Fabrication: the co-evolution of form and formation.* PhD thesis, Brandeis University, 2006.
9. John Rieffel and Jordan Pollack. The Emergence of Ontogenic Scaffolding in a Stochastic Development Environment. In Kalyanmoy Deb et al., editor, *Proceedings of the 2004 Genetic and Evolutionary Computation Conference (GECCO-2004)*, pages 804–815, Seattle, Washington, USA, June 2004. Springer-Verlag, Lecture Notes in Computer Science Vol. 3102.
10. John Rieffel, Francisco Valero-Cuevas, and Hod Lipson. Automated discovery and optimization of large irregular tensegrity structures. *Computers & Structures*, 87(5-6):368 – 379, 2009.
11. Karl Sims. Interactive evolution of dynamical systems. In *First European Conference on Artificial Life*. MIT Press, 1991.
12. Karl Sims. Evolving 3d morphology and behavior by competition. In R. Brooks and P. Maes, editors, *Artificial Life IV Proceedings*, pages 28–39. MIT Press, 1994.
13. Adrian Thompson. Silicon evolution. In *Stanford University*, pages 444–452. MIT Press, 1996.
14. T. Vilbrandt, E. Malone, H. Lipson, and A. Pasko. Universal desktop fabrication. *Heterogenous Objects Modeling and Applications*, pages 259–284, 2008.
15. Richard A. Watson, Sevan G. Ficici, and Jordan B. Pollack. Embodied evolution: Embodying an evolutionary algorithm in a population of robots. In Peter J. Angeline, Zbyszek Michalewicz, Marc Schoenauer, Xin Yao, and Ali Zalzala, editors, *Proceedings of the Congress on Evolutionary Computation*, volume 1, pages 335–342, Mayflower Hotel, Washington D.C., USA, 6-9 1999. IEEE Press.
16. D. Whitley, J. R. Beveridge, C. Guerra-Salcedo, and C. Graves. Messy genetic algorithms for subset feature selection. In *International Conference on Genetic Algorithms, ICGA-97*, 1997.