Artificial Ontogenies for Real World Design and Assembly

John Rieffel[1], Jordan Pollack[1]

[1] DEMO Lab, Brandeis University, Waltham MA, 02454, USA
jrieffel@cs.brandeis.edu

Abstract

Relatively few evolved designs have made the transition to the real world. Of those that have, all have been built by hand based upon descriptive representations (i.e. blueprints) of the evolved object. As such, human effort is transferred from the design to the assembly domain. In this paper we suggest harnessing the procedural representations provided by Artificial Ontogenies to fully automate both design *and* assembly. We demonstrate the ability of Artificial Ontogenies to cross one hurdle of real-world assembly, namely reliably building structures in noisy environments. We then discuss the advantages of Artificial Ontogenies for Automated Design and Assembly, and offer suggestions for the future of the field.

Introduction

Despite the popularity of Evolutionary Design, relatively few designs have actually made the transition from simulated environments to the real world. In a brief review, below, of those that have made the transition, the most successful have been based upon Artificial Ontogenies (Kumar and Bentley, 2003; Stanley and Miikkulainen, 2002), which rely upon indirect representations of the evolved object. Yet while each of the objects in our review were automatically designed via evolution, each was subsequently built by hand, based upon a *direct* representation of the evolved object. Therefore, while human effort has been removed from the realm of design , it has been correspondingly increased in the realm on assembly.

In this paper we would like to suggest that by properly harnessing the demonstrated utility of indirect representation that is provided by Artificial Ontogenies, evolved designs can be not just automatically designed, but automatically *assembled* as well, thereby fully removing the human from the loop.

Obvious barriers to fully automated design and assembly exist, most notably the so-called *reality gap* between simulated environments and the real world. In particular, the stochasticity of real-world environments imposes quite a strain on automated assembly. Fortunately, as we have recently demonstrated (Rieffel and Pollack, 2004), although Artificial Ontogenies can be brittle in the face of noisy development environments, they are also able to develop robust solutions in the face of such noise – by means of *ontogenic scaffolding*.

The purpose of this paper is to explore the strengths and weaknesses of of Artificial Ontogenies for real-world design and assembly. We first provide a review of evolved designs which have made the transition to the real world. We then discuss the advantages of using Artificial Ontogenies for design, describe our recent work, and provide new analysis of our earlier results. Finally, we discuss the potential of Artificial Ontogenies for fully automating both design and assembly.

Review of Real-World Evolved Designs

There have been quite a few impressive results in Evolutionary Design, starting perhaps with Sims' (Sims, 1994) seminal work. Despite this, very few evolved designs have been transferred to the real world.

Of those that have made the transition, the most notable early example is Funes' LEGO structures (Funes, 2001). In this work, structures were evolved in a simulator which took into account forces between bricks. Successful results were then built by hand. Subsequently, Lipson (Pollack et al., 2001) evolved mobile robots (Golems) in a simulator. Successful robots were transferred by hand into CAD designs and printed on a 3-D rapid protyping machine. Motors, wires, and circuits were added by hand. Both Funes and Lipson used direct representations of their structures.

Hornby (Pollack et al., 2001) used L-systems, a type of Artificial Ontogeny, to evolve tables and mobile robots. The developed genotype consisted of instructions to a LOGO-like turtle which then "drew" the structures out of voxels in simulation. Although early results were transferred by hand into CAD before printing on a 3-D printer, Hornby's later designs created CAD files automatically. It is worth noting, however, that CAD designs are *descriptive*, and have to be translated by the 3-D printer into specific instructions for the printing apparatus. Like Lipson's Golems, once the bodies were printed, final assembly, including the addition motors

Table 1: A Review of Evolved Designs in the Real World

	Genotype	Final Design	Assembly
Funes	Direct	Descriptive	Hand
Lipson	Direct	Descriptive	Semi-Auton.
Hornby	Indirect	Descriptive	Semi-Auton
Lohn et al	Indirect	Descriptive	Hand

and wiring, was performed by hand.

The most significant recent result of real-world evolved design is probably Lohn et al's work on Evolved Antennas (Lohn et al., 2003) - one of which at least is due to be launched into space aboard a Low Earth Orbit satellite. These designs were generated by L-systems in a manner similar to Hornby's work, and tested in an antenna simulator. Successful antenna designs were represented descriptively, and then meticulously built by hand.

Table 1 provides a comparison of these designs and the methods used. Of those reviewed, regardless of whether they used direct or indirect encodings as their genotypes, when it came time for assembly of successful designs, they were build by hand from descriptive representations. Lipson's and Hornby's both used rapid prototyping machines to semi-autonomously fabricate parts, but final assembly was performed by hand.

Artificial Ontogenies for Real World Design and Assembly

Unlike traditional evolutionary computation, Artificial Embryogenies (Stanley and Miikkulainen, 2002; Kumar and Bentley, 2003) treat the genotype as an *indirect*, or *procedural* encoding of the phenotype. The genotype is decoded and transformed into a phenotype by means of some developmental process. This abstraction layer between genotype and phenotype allows for quite a bit of flexibility during evolution, and has several demonstrated advantages (Hornby and Pollack, 2001; Toussaint, 2003; Stanley and Miikkulainen, 2002; Kumar and Bentley, 2003; Bongard and Pfeifer, 2001).

Assembly Plans as Artificial Ontogenies

Most of the real-world evolved designs reviewed above were the result of GAs which generated as an end product a *blueprint* of the final design. Blueprints are descriptive representations of structure – as such, they contain no information about how to actually build the goal structure. In fact, in the systems above, significant human knowledge and interaction was required to translate the blueprints into physical objects. Thus, while the evolution of blueprints removes human effort from the design task, it fails to remove human effort from the assembly task - and may in fact increase it.

Artificial Ontogenies on the other hand, because they are prescriptive, rather than descriptive representations, can provide step-by-step instructions on how to build an evolved design. A particular form of Artificial Ontogeny that we are interested in is the *assembly plan*. We define an assembly plan as a linear, ballistic[1] , set of instructions to an external builder which when executed results in the construction of a structure. As such, assembly plans allow for the *full automation* of evolutionary design and assembly, provided of course that the assembly mechanism can interpret and execute the instructions contained within the assembly plan.

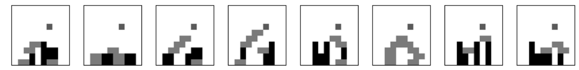


Figure 1: A sample of the distribution of phenotypes when a single assembly plan is built in a noisy environment

There are two clear advantages to fully automated assembly. The first is the simple fact that in many contexts, such as space exploration, it is much less expensive and much safer to send fully automated assembly plants than it is to send human labor. Secondly, machine fabrication offers significantly higher precision and reliability than human assembly - while this difference may not be noticeable on relatively simple tasks such as wire-bending for antennas, as evolved designs become more complex, and their behavior more finely tuned and nuanced, the need for precision and reliability will increase considerably.

The Effects of Noise during Development

As we show in our earlier work (Rieffel and Pollack, 2004), despite their strengths, linear ballistic assembly plans have a clear weakness - that of noise during development which can lead to errors in the final assembly. Because each step of the assembly can be predicated upon the success of earlier steps, a single failure, particularly early in the assembly, can have severe consequences on the final outcome.

Furthermore, in the context of evolution, noise during development results in each genotype developing into an entire distribution of phenotypes, each with a corresponding set of fitness values, rather than one single phenotype, as shown in Figure 1. This range of fitness values introduces a credit assignment problem - which, if any, of the resulting phenotypes can be considered representative of the single source genotype?

Emergence of Ontogenic Scaffolding

We have been able to demonstrate, however, that by incorporating noise into the development environment used during evolution, Artificial Ontogenies are able to overcome noisy development and reliably build a goal structure by means of *ontogenic scaffolding* - intermediate structural elements that

[1]That is to say, without any ability to test intermediate results, or alter their behavior mid-assembly

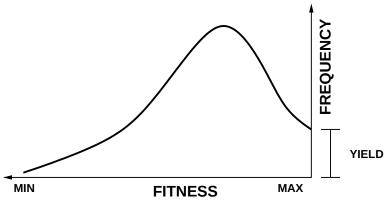


Figure 2: A noisy development environment leads to a distribution of phenotypic fitnesses. Yield is the frequency with which the distribution reaches the maximum fitness

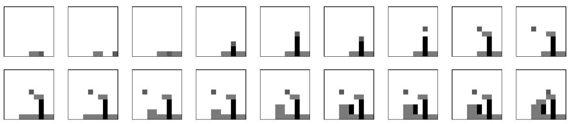


Figure 3: Robust Assembly Plan Steps 1-18: In the first steps, the builder lays scaffolding (frames are read left to right, top to bottom)

are necessary for assembly of, but are not present in, a final structure.

Our experiments used a simple 2-D grid environment with very simple physics and a LOGO-like turtle, capable of placing 2x1 bricks, as the interpreter of the evolved assembly plans. A noisy development environment was induced by allowing vertical bricks to topple to either side 50% of the time, and for cantilevered bricks to topple 50% of the time. Our genotype assembly plans contained instructions such as forward, rotate, put brick and take brick. In order to account for the distribution of phenotypes, each genotype was built 50 times in the noisy environment, and statistical properties of the results used as fitness objectives.

The salient fitness measure used was that of *yield*. In the case where there is an achievable maximum fitness, yield corresponds to the frequency with which the maximum fitness is attained (see Figure 2. In our experiments, in which solutions attempted to build a goal structure, yield is the percentage of times that a genotype was able to perfectly build the goal.

The details of the experiments are provided in our earlier paper (Rieffel and Pollack, 2004). Our core result is shown in Figures 3 thru 5, which show frames of a resulting phenotype able to reliably build the goal arch 75% of the time. It is worth noting that the evolved solution contains two distinct phases of ontogeny. In the first, the scaffolding is laid and the structure is build within the scaffolding. In the second, the scaffolding is removed, leaving only the final structure. This suggests that the nature of the solution, not just the name of the method, is an ontogeny.

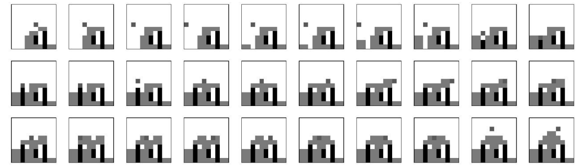


Figure 4: Robust Assembly Plan Frames 19-49: more scaffolding is lain and the arch is completed

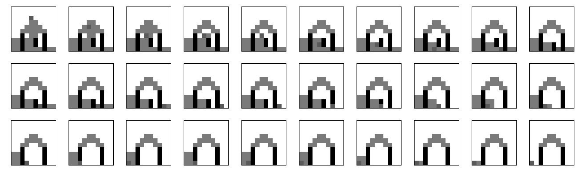


Figure 5: Robust Assembly Plan Frames 50-80: scaffolding is removed

Subsequent Analysis

A meaningful way to visualize results which contain a distribution of phenotypes corresponding to a single genotype is to build a composite image by averaging the results of multiple runs of that genotype. The center column of Table 2 compares composites of a naive solution (which was evolved in the absence of noisy assembly and then assembled under noise) with evolved solutions which achieved 11%, 27% and 59% and 75% yield. As can be seen, as evolution progresses, the composite image increasingly looks like the final goal structure.

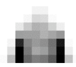
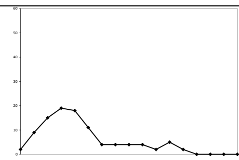
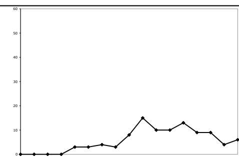
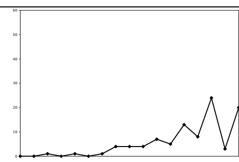
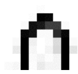
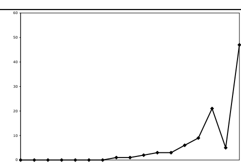
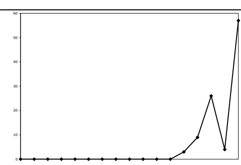
In the context of noisy development in which genotypes are rewarded for their yield percentage of a final structure, such as ours, one can consider the role of evolution as learning to shift phenotypic fitness distributions, rather than individual fitnesses, towards the optimal. This is borne out by the distribution column in Table 2. Each figure is a histogram which shows the distribution of phenotype fitness over the same 100 builds used to generate the composite images. As shown, as the yield increases, the distribution tightens and shifts towards the optimal.

Discussion

Artificial Ontogenies have great potential for fully automated design and assembly. In terms of automated design aspect, they have many demonstrated advantages, such as the ability for co-ordinated parallel changes to the phenotype (Hornby and Pollack, 2001), the ability to switch representations via neutral mutations (Toussaint, 2003), and the many-to-one genotype-phenotype mapping (Stanley and Miikkulainen, 2002; Kumar and Bentley, 2003).

Their potential for automated assembly lies in their ability to procedurally describe each step of an object's assembly. As we've shown, this latter ability is particularly useful in

Table 2: The middle column contains composite results created by averaging 100 builds - darker squares represent locations more likely to contain a brick. The right hand column shows the histogram distribution of phenotype fitness across those same 100 builds - the horizontal axis represents increasing fitness, with maximal fitness, meaning perfect assembly, on the extreme right. Scales between histograms are identical

Result	Composite	Distribution
Naive		
11%		
27%		
39%		
75%		

noisy environments, where Artificial Ontogenies are able to develop scaffolding - intermediate structural elements necessary for the reliable assembly of a final object (Rieffel and Pollack, 2004) Most importantly perhaps, the procedural nature of Artificial Ontogenies allows for them to be interpreted and executed by a machine, without human intervention.

We would like to suggest that a crucial requirement for this full automation is that *the language of development MUST be the same as the language of assembly*. That is to say, the step-by-step instructions contained in the genotype-as-assembly plan used for development must also be interpretable by the agent ultimately responsible for building the design in the real world. Without this, the language of development has to be translated into a language of assembly, and all of the pitfalls of the reality gap between simulation and reality re-emerge. Therefore, by evolving designs using artificial ontogenies, by using simulated development environments which sufficiently represent the noise of the final real-world assembly environment, and by ensuring that the languages of development and assembly are identical, we hope to achieve the goal of fully automated design and assembly.

References

Bongard, J. C. and Pfeifer, R. (2001). Repeated structure and dissociation of genotypic and phenotypic complexity in artificial ontogeny. In Spector, L. et al., editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 829–836, San Francisco, California, USA. Morgan Kaufmann.

Funes, P. (2001). *Evolution of Complexity in Real-World Domains*. PhD thesis, Brandeis University, Dept. of Computer Science, Boston, MA, USA.

Hornby, G. S. and Pollack, J. B. (2001). 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. IEEE Press.

Kumar, S. and Bentley, P. J. (2003). *Advances in evolutionary computing: theory and applications*, chapter Computational embryology: past, present and future, pages 461–477. Springer-Verlag New York, Inc.

Lohn, J. et al. (2003). Evolvable systems for space applications. In *International Conference on Space Mission Challenges for Information Technology (SMC-IT)*.

Pollack, J. B., Lipson, H., Hornby, G., and Funes, P. (2001). Three generations of automatically designed robots. *Artifial Life*, 7(3):215–223.

Rieffel, J. and Pollack, J. B. (2004). The emergence of ontogenic scaffolding in a stochastic development environment. In Deb et al., editors, *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-2004*. Springer-Verlag.

Sims, K. (1994). Evolving virtual creatures. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 15–22. ACM Press.

Stanley, K. O. and Miikkulainen, R. (2002). A taxonomy for articial embryogeny. *Artificial Life*, 9(2):93–130.

Toussaint, M. (2003). Demonstrating the evolution of complex genetic representations: An evolution of artificial plants. In *Proceedings of the 2003 Genetic and Evolutionary Computation Conference (GECCO 2003)*. Springer-Verlag, New York.