On Attaining Maximal Fitness or How to Evolve Virtual Redwoods

Gary R. Greenfield

Mathematics & Computer Science
University of Richmond, Richmond, VA 23173
ggreenfi@richmond.edu

Abstract

An understanding of the evolutionary mechanisms that lead to increased genome capabilities is essential for determining the potentials and limits of living systems. We evolve abstract aesthetic imagery on the basis of an image connectivity trait that has a theoretically determined maximum value. This trait can be influenced by "development" genes. We investigate simulated evolution when the development genes are not present, when they are fused with the existing genotype, and when they function semi-autonomously by residing in symbionts. Genome fusing initially impedes evolution while symbiosis initially accelerates evolution. Our findings suggest that symbiosis events play a more important role than genome fusion events for sustaining open-ended evolution.

Keywords: Evolving expressions, computational aesthetics, simulated evolution.

Introduction

We present evidence favoring genome "cooperation" over genome "fusion" as a mechanism for enhancing the survival of a species. Our results are based on simulated evolution experiments using a species that possesses a trait that is easy to formulate, has a theoretically determined maximum fitness value, and can be suitably influenced by genes. The genotypes for our species are postfix expressions and the phenotypes are abstract images. Image phenotypes undergo a development phase, after which they are evaluated for a "connectivity" trait. The development phase can be partially regulated by coefficient genome vectors. Our design allows these vectors either to be suppressed, to be fused with image genotype expressions, or to function semi-autonomously, as primitive, rapidly evolving symbionts.

If coefficient genome vectors are suppressed, then a randomly generated population of image genotypes will settle upon a genome "template" from which it will either successfully attain the maximum value of the trait within 250 of the 500 allotted generations or become trapped in an evolutionary cul-de-sac. If coefficient genome vectors are fused with image genotypes, then randomly generated populations support a wider range of genome templates to start with and provide a larger fitness landscape to explore, whence image populations maintain better *average* fitness and evolution proceeds slowly and inexorably along an evolutionary trajectory towards the maximum value of the trait. If a hill-climbing algorithm allowing coefficient genome vectors to function as independent, fast evolving subspecies of symbionts is used, then randomly generated image populations can either accelerate rapidly to the maximum value of the trait or evolve steadily towards the maximum value. These results support the claim that symbiosis events are more important than genome-fusion events for sustaining open-ended evolution. Moreover, they suggest that genome fusion is an evolutionary neutral event, contra-indicated when the traits affected by the fused genes are under immediate evolutionary pressure.

Background and Motivation

"User-guided evolution" requires a user to assign fitness rankings to the phenotypes in each generation in order to determine the breeding population for the next generation. This paradigm is cental to the genre of computer generated art referred to as evolutionary art or generative art. Difficulties arise when phenotype rendering times, the number of phenotypes to examine, or the number of generations needed to evolve acceptable aesthetics become too large. In such circumstances it becomes desirable to automate the phenotype acceptance/rejection process. When this is done, we say evolution is driven by *computational* aesthetics. Absolute measures of aesthetic fitness are usually a deterrent to open-ended evolution (Baluja et al 1994), therefore coevolutionary methods based on competition between two species (Greenfield 2000b)) or cooperation between two species (Greenfield 2002)) have been considered. In the latter study, aesthetic fitness based on geometric data obtained by color segmenting aesthetic image phenotypes was investigated. Color segmenting decomposed each image into simply-connected regions whose areas, boundary lengths, and number of region adjacencies could be calculated. In one experiment, the total number of region adjacencies

was used as the fitness criterion to try and evolve images whose "compositions" exhibited multiple thread-like shapes. Surprisingly, compact, minimalist compositions were obtained (see Fig. 1.) The explanation is

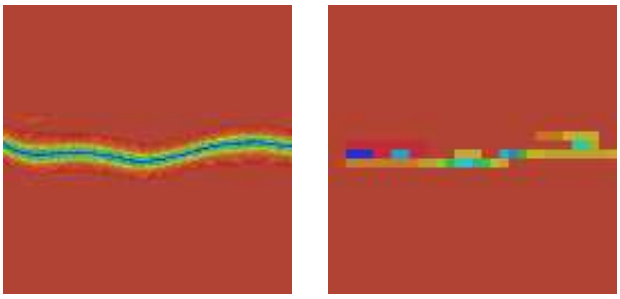

Figure 1: Left: Full resolution 128×128 color image evolved using computational aesthetics. Right: Color segmented 32×32 image used to determine aesthetic fitness.

that the *graph* obtained by interpreting regions as vertices and adjacencies as edges is always a simple, connected, planar graph and there is a theoretical limit to the number of edges in a planar graph on n vertices, namely $3n-6$ (Behzad et al 1979), which is significantly smaller than the upper bound of $n(n-1)/2$ one might expect. The most fit images produced using these methods are long-lived organisms belonging to a species with few surviving specimens, capable of producing few, if any, breeding offspring. We are reminded of species such as the giant redwood (*Sequoia gigantea*) or bristlecone pine (*Pinus longaeva*).

The process of organizing a digital image into a prescribed number of simply-connected regions is the *development* phase a newly bred image phenotype undergoes before reaching maturity. Parameters, or genes, for controlling this development can be used to help formulate three types of simulated evolution: (1) evolution using fixed values for these genes so that development is determined by an environmental litmus test, (2) evolution where genes for regulating the environment are acquired by the species in a genome fusion event, and (3) evolution where genes for regulating the environment reside in a cooperating (i.e., symbiotic) species.

Color Images from Evolving Expressions

Let $\{c_1, \ldots, c_L\}$ be a set of colors. We let $L = 450$. Each color $c_k = (h_k, s_k, v_k)$ has hue $h_k \in [0, 6]$, saturation $s_k \in [0, 1]$, and value $v_k \in [0, 1]$. A function $F : [0, 1] \times [0, 1] \longrightarrow [0, 1)$ generates an $m \times m$ color image by assigning pixel $p_{i,j}$, where $0 \le i, j < m$, color c_k if and only if $F(i/m, j/m) \in [(k-1)/L, k/L)$. The method of evolving expressions constructs an evolvable class of such functions from a set of *primitives* (Sims 1991). Image genotypes are postfix expressions built from such primitives. The primitives we use (Greenfield 2000a) include variables V_0 and V_1; constants $C_0, C_1, \ldots, C_{999}$; unary functions $U_0, U_1, \ldots, U_4$; and binary functions $B_0, B_1, \ldots, B_{14}$. An example postfix expression is:

```
V1 V1 V0 B6 V1 B4 V1 V1 V0 B6 B0 V0 V1 B9 V0
B1 B13 V0 V1 B1 V1 B14 V1 B4 V0 B5 V0 B11 V0
B1 B1 V1 V0 B11 V0 B3 V1 V0 B4 V0 B1 B2 B1
B14 B9
```

Image phenotypes are generated by setting $V_0 = i/m$, $V_1 = j/m$, and performing a postfix evaluation. The recombination operator is subtree crossover. The mutation operator is "bit-flipping" which replaces, with probability p_{mut}, each primitive by a new one of the same arity.

Segmentation of Color Images

To color segment, we render image phenotypes at a resolution of 32×32 and use each pixel to initialize a region of area one whose averaged color matches the pixel's color. Using a two-stage process, we perform a sequence of merges of adjacent regions until only 25 regions remain. The first stage, priority-merging, iteratively selects and merges the two adjacent regions whose average color is "closest" until the number of regions is reduced to 50. The second stage, absorption-merging, iteratively selects and merges the smallest remaining region with its largest adjacent neighbor until the number of regions is reduced to 25. Color "closeness" loses its meaning in darker areas so excessive priority-merging can "corrupt" an image. Moreover, since region adjacencies are determined from pixel *edges*, diagonal cascades of regions can disrupt the formation of adjacencies. Absorption-merging helps removes diagonal cascades (see Fig. 2.)

To find the adjacent regions whose averaged colors are "closest" we must determine the boundary edge of minimal priority. Let (h_1, v_1, s_1) and (h_2, v_2, s_2) be the averaged colors of regions R_1 and R_2 bounded by edge e. Set $\Delta_h = \min(|h_1 - h_2|, 6 - |h_1 - h_2|)$, $\Delta_s = |s_1 - s_2|$, and $\Delta_v = |v_1 - v_2|$. Define the edge priority $p(e)$ to be

$$\begin{aligned} p(e) \quad = \quad & (1 + k_{h,s}\Delta_s + k_{h,v}\Delta_v + k_{h,v,v}\Delta_v^2 \\ & + k_{h,v,s}\Delta_v\Delta_s + k_{h,s,s}\Delta_s^2)\Delta_h \\ & + k_s\Delta_s + k_v\Delta_v. \end{aligned}$$

The coefficient genome vector is the 7-tuple

$$(k_{h,v}, k_{h,v,s}, k_{h,s}, k_v, k_s, k_{h,v,v}, k_{h,s,s}),$$

where each coefficient lies in the interval $[0, 3]$. Assuming $h_1 \ge h_2$, the merged region is assigned averaged color $(\bar{h}, \bar{s}, \bar{v})$ where $\bar{s}$ and $\bar{v}$ are the usual area weighted averages, but due to the circular nature of the hue scale its area weighted average is given by $\bar{h} = (a_1 h_1 + a_2(b + h_2))/(a_1 + a_2) \pmod 6$, with $b = 0$ if $h_1 - h_2 \le 3$ and $b = 6$ otherwise. The recombination operator is one-point crossover and the mutation operator

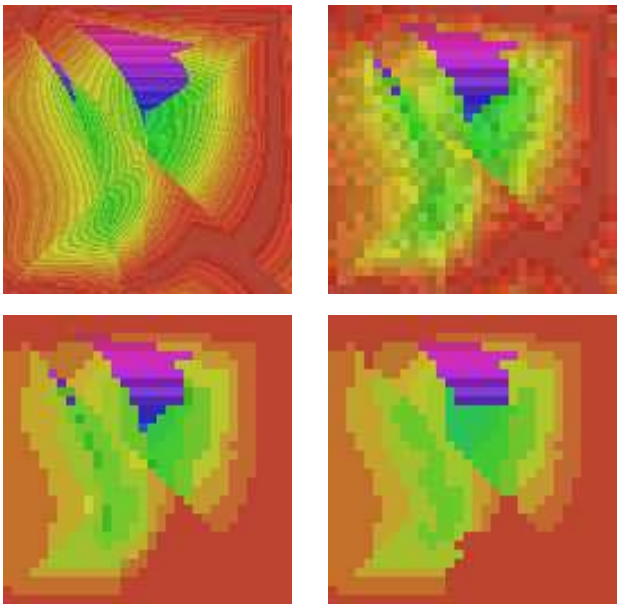

Figure 2: Top Left: High resolution 128×128 image. Top Right: Low resolution 32×32 image. Bottom Left: Low resolution image color segmented to 50 regions by priority-merging. Bottom right: Low resolution image further color segmented to 25 regions by absorption-merging.

is coefficient perturbation. The coefficient genome used to color segment the image phenotype in Figure 2 was $(3.0, 0.0, 0.0982277, 1.22504, 0.0592963, 1.32771, 2.37181)$.

Planar Graphs and Image Fitness

To construct a graph from a color segmented image, first label the pixels within each color segmented region using unique vertex labels to obtain a 32×32 connectivity *tableau*, then use the tableau to construct a 25×25 adjacency matrix. Fig. 3 shows the top portion of the tableau produced from the example in Fig. 2 by using vertex labels `a` through `y`, while Fig. 4 shows the first few rows of the adjacency matrix obtained from this tableau. Image fitness is the sum of the vertex adjacencies divided by two. For our example, this fitness is $136/2 = 68$, one shy of the maximum.

Experimental Design

Image population size is set to 30. After each image generation, the 12 most fit individuals provide replacements for the 18 least fit individuals by breeding 9 randomly selected pairs (with replacement). Since there is no "reaper" in the sense of (Ray 1992), and no provision for marginally fit images to survive to breed, with probability p_{sev}, a member of a breeding pair can be replaced with a randomly generated image genotype. The three types of simulated evolution we implemented were:

```
yyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
yyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
yyyyyyyyyyyyppppppppppyyyyyyyy
yylyyykkkkccccccmmmmmppppppyyyy
lllrykkkkkgooooooooommmmpppyyyy
lllrykkkkgghooooooottmmmmpppyyy
lllrrrkkgghheeeeeeeettttmppyyyy
lllrrwwgghhvvddddddtttttmppyyyy
llrrrwwwghhvvxbbbbbbtnnttmppyyyy
llrrrwwwuhhvvxaaaaaatnnttmppyyyy
```

Figure 3: The top ten rows of the 32×32 connectivity tableau constructed from the example image phenotype following the development phase.

```
  abcdefghijklmnopqrstuvwxy
a 01000000100000000001100010   5
b 10010000000000000000100010   4
c 00000010001010110000000001   6
d 01001000000000000000101010   5
e 00010001000000100000101000   5
f 00000000010001000010000001   4
g 00100001001000010000010100   6
h 00001010000000100000011000   5
```

Figure 4: The first eight rows of the 25×25 adjacency matrix constructed from the connectivity tableau of our example. Vertex adjacency counts are shown on the right.

N-Genes. An organism genotype consists of an image genotype. The coefficients of the edge priority function are all set to zero, except for k_s and k_v which are set to one.

F-Genes. An organism genotype consists of an image genotype *fused* with a coefficient genotype. When images are bred, 75% of the time a newborn inherits its coefficient genotype from one of its parents, and 25% of the time it inherits a coefficient genotype resulting from the crossover of the coefficient genotypes of its parents.

S-Genes. An organism genotype consists of an image genome *partnered* with a coefficient genome. In between each *image* generation, for 20 *coefficient* generations, coefficient genomes are evolved asexually, and the coefficient genotype for the organism is replaced if and only if the fitness of its image phenotype will be improved. (This hill-climbing algorithm is an evolutionary algorithm that applies a separate local search process to refine individuals. It provides the illusion that each coefficient genotype is a canonical representative for a *population* of symbionts.) When images are bred, the 75:25 scheme for coefficient genotypes remains in effect.

Simulation Results

When genes for regulating development were suppressed, the average fitness, taken over 20 trials, of the most fit individual in the initial randomly generated population was 52.650. When such genes were present, the average fitness, taken over 40 trials, of the most fit individual in the initial population was 53.625. This shows there was no evolutionary bias due to randomization. After 100 image generations, the average fitness of the most fit individual, taken over 20 trials, was 65.75 for F-Genes organisms, 66.35 for N-Genes organisms, and 67.5 for S-Genes organisms. This shows that fitness transients had died out by this time, and helps support the argument that genome fusion events initially *slow* the rate of fitness improvement. However, after 500 image generations the average fitness for the most fit individual in the F-Genes case was 68.4, overtaking the average fitness in the N-Genes genes case which was 68.1.

S-Genes trials were only run for 100 generations, so in order to compare evolution rates, we considered only those trials where the most fit individual had obtained a fitness of at least 65 by that point. Table 1 dramatically reveals that genome fusion slowed evolution while symbiont genomes accelerated evolution. N-Genes and F-Genes trials were run for 500 generations. For 20 trials of each type, it took an average of 67.75 generations for the top individual to reach fitness 65 for N-Genes trials, and 76.25 generations for F-Genes trials.

Type	Trials	Ave. No. Gens.
N-Genes	17	52.35
F-Genes	15	61.53
S-Genes	18	37.22

Table 1: The average number of generations that it took for the most fit individual to reach fitness 65. The time limit was 100 image generations.

Our objective was to evolve individuals that attained the maximum fitness value of 69. Fig. 5 shows an example of a "perfect" specimen, our virtual redwood tree. Table 2 shows how successful we were in achieving our objective within the first 100 generations. It reveals the clear advantage symbionts provided. In 20 trials of the N-Genes simulation the fitness maximum was achieved 10 times within 500 generations, and in 20 trials of the F-Genes simulation the fitness maximum was achieved 11 times within 500 generations. Even though the probability for success was virtually identical, the average number of generations required to attain this limit soared from 167.3 in the N-Genes case to 288.4 in the F-Genes case. More striking is the fact that only twice during N-Genes trials was evolution able to attain the maximum after 250 generations, while it was attained seven times after 250 generations during F-Genes trials. Thus, when no genes for regulating development were present, either maximal fitness was reached very quickly or evolution got stuck.

Type	Trials	Successes	Ave. No. Gens.
N-Genes	20	3	85.0
F-Genes	20	2	74.5
S-Genes	20	6	46.8

Table 2: The number of trials which successfully attained the fitness limit within 100 generations together with the average number of generations it took when this occurred.

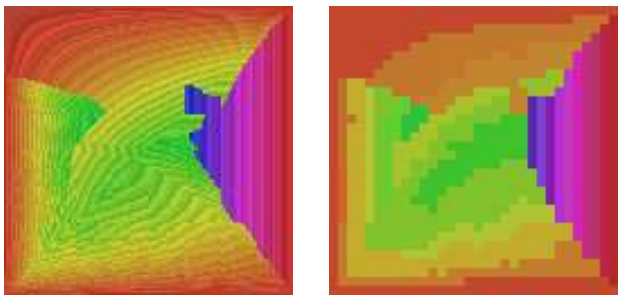

Figure 5: An maximally fit S-Genes example. Left: 128×128 resolution. Right: Color segmented image.

This work conducted while on sabbatical at the Visualization Laboratory at Texas A&M University.

References

Baluja, S.; Pommerleau, D.; and Jochem, T. 1994. Towards automated artificial evolution for computer-generated images. *Connection Science* 6:325–354.

Behzad, M.; Chartrans, G.; and Lesniak-Foster L. 1979. *Graphs & Digraphs*. Belmont, CA: Wadsworth.

Greenfield, G. 2000a. Mathematical building blocks for evolving expressions. In Sarhangi, R., ed., *2000 Bridges Conference Proceedings*, 61–70. Winfield, KS: Central Plain Book Manufacturing.

Greenfield, G. 2000b. Art and artificial life — a coevolutionary approach. In Bedau, M. et al, eds., *Artificial Live VII Conference Proceedings*, 529–536. Cambridge, MA: MIT Press.

Greenfield, G. 2002. Color dependent computational aesthetics for evolving expressions. *2002 Bridges Conference Proceedings*, to appear.

Ray, T. 1992. An approach to the synthesis of life. In Langton, C., et al, eds., *Artificial Life II, SFI Studies in the Science of Complexity*, 372–405. Reading, MA: Addison-Wesley.

Sims, K. 1991. Artificial evolution for computer graphics. *Computer Graphics*, 25:319–332.