SOUTHWESTERN UNIVERSITY

BROWN WORKING PAPERS IN THE ARTS & SCIENCES

Volume VII

(2007)

A Tasteful Example of Evolutionary Programming

Christy Bell
Department of Chemistry
Southwestern University
Georgetown, TX 78626

Steven Alexander
Associate Professor of Physics
Department of Physics
Southwestern University
Georgetown, TX 78627
alexands@southwestern.edu

Recommended Citation:

Bell, Christy and Steven Alexander. (2007) "A Tasteful Example of Evolutionary Programming," Brown Working Papers in the Arts & Sciences, Southwestern University, Vol. VII. Available at: http://www.southwestern.edu/academic/bwp/vol7/bellalexander-vol7.pdf.

A Tasteful Example of Evolutionary Programming

Christy Bell
Department of Chemistry
Southwestern University
Georgetown, TX 78626

S.A. Alexander
Department of Physics
Southwestern University
Georgetown, TX 78626

ABSTRACT

We describe a simple algorithm that has been used to determine the optimum mixture of 15 liquids. The fitness function in this project is the subjective judgement of a taster. The convergence of our algorithm is sufficiently fast that it can be used as a demonstration of evolutionary programming.

INTRODUCTION

Evolutionary programming is an optimization strategy that is based on the stochastic modification of a set of trial solutions [1]. This technique is especially well suited for combinatorial problems and situations where the fitness landscape has many local minima. Because of their flexibility, evolutionary algorithms are frequently used to construct aesthetic evolutionary design systems. These software tools allow non-expert designers to “discover” interesting design solutions via an exploration-based, rather than construction-based, interface.

One of the first aesthetic evolutionary design systems to be created was Dawkins' biologically inspired Biomorphs program that evolved 2D insect-like drawings [2]. In 1991 Karl Sims used this technique to create computer graphics objects from a combination of mathematical functions [3]. In this and subsequent evolutionary art programs, an initial population of randomly generated objects is created. The artist then selects one or more parent pictures based on their aesthetic characteristics. These "parents" are mutated and/or crossbred to produce a number of "children" that will form the starting point for the next iteration. The role of the artist in this process has frequently been compared to a gardener who steers evolution by breeding and weeding plants [4]. In such a situation survival of the "fittest" is replaced by survival of the most "aesthetic".

In recent years aesthetic evolutionary design systems have been used to create a variety of images [4,5], sculpture [6], architecture [7,8], consumer products [9,10] and several types of music [11,12]. In this paper we use these same techniques to find the optimum mixture of 15 liquids. The fitness function in this example is the subjective

judgement of a taster. In section 2 we describe our algorithm and in section 3 we describe our results. To the best of our knowledge this is the first time that evolutionary programming has been applied to the sense of taste and we believe that it provides an interesting demonstration of this technique.

ALGORITHM

Our algorithm, like most evolutionary programs, consists of three parts:

(1) Generate an initial population of trial solutions. For this project we selected a collection of 15 liquids that represent a wide assortment of common flavors. These liquids are listed in Table 1. We generated our initial solution vector by randomly selecting 15 values between zero and one. Next we normalized each value by dividing it by the sum of all the values in the vector. After this step each value now represents the percentage of that liquid in the mixture. Finally we generated nine additional vectors by randomly perturbing each element in the initial vector and then normalizing the results.

Table 1. The liquids used in our project

Sugar water	Ginger flavoring	Strawberry nectar
Salt water	Pineapple juice	Tea
Raspberry juice	Peach nectar	Milk
Grapefruit juice	Cranberry juice	Coffee
Apple juice	Grape juice	Chocolate

(2) Evaluate the fitness of each solution. In order to form one cup of a given mixture we multiplied the values in that vector by 48 teaspoons. That amount of liquid was then placed in a container. After sampling all ten mixtures the taster picked the top three based solely on her subjective judgement. These became the parents of the next generation of solution vectors unless a maximum number of iterations was reached or an adequate solution was obtained.

(3) Mutate the parents according to a statistical distribution that weights small changes as highly probable and substantial changes as increasingly unlikely. Our mutation algorithm is modeled after the one described by Todd and Latham [4]. We begin by designating the top three choices in a generation as the vectors a1(i), a2(i) and a3(i). Similarly the top choice from the previous generation is defined as a0(i). From these four vectors we create three direction vectors that will help to steer new solutions toward more acceptable regions

$$\Delta 1(i) = a1(i) - a0(i)$$
$$\Delta 2(i) = a2(i) - a0(i)$$
$$\Delta 3(i) = a3(i) - a0(i)$$

The ten new solution vectors that form the next generation can now be defined as

$$v1(i) = a1(i)$$
$$v2(i) = a1(i) + \Delta 1(i)$$
$$v3(i) = a1(i) + 2\Delta 1(i)$$
$$v4(i) = a1(i) + \Delta 1(i) + r1(i)$$
$$v5(i) = a1(i) + \Delta 1(i) + r2(i)$$
$$v6(i) = a1(i) + \Delta 2(i)$$
$$v7(i) = a1(i) + 2\Delta 2(i)$$
$$v8(i) = a1(i) + \Delta 2(i) + r3(i)$$
$$v9(i) = a1(i) + \Delta 2(i) + r4(i)$$
$$v10(i) = a1(i) + \Delta 3(i)$$

where rn(i) is vector containing a random perturbation in the range [-1,1]. At the end of this step each new solution vector is normalized. If any of the direction vectors are zero, the new vector is simply a randomly perturbation of each element in a0. By including the top choice from the previous generation in the current generation we hope to provide a reference from which a reliable decision can be made.

To help assess the convergence of our algorithm we computed the sum of the absolute differences between the current top choice and the previous top choice

$$\Sigma \mid a1(i) - a0(i) \mid$$

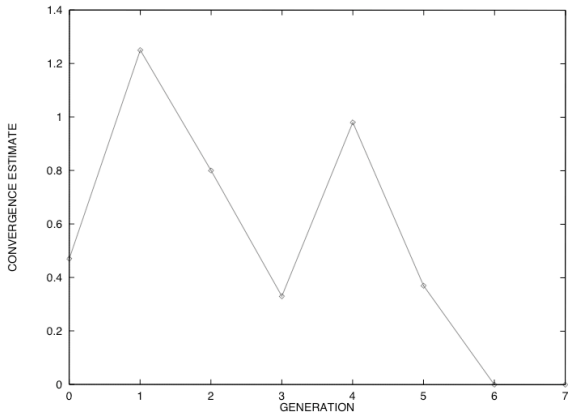
Although this expression provides no information about how a particular mixture tastes, it does measure how much the percentages of each liquid are changing in the top choice from one generation to the next.

RESULTS

As might be expected from a set of randomly mixed liquids, all of the drinks in our first generation tasted revolting. All that the taster could do in this situation was choose the lesser of ten evils. Many of the mixtures in the second generation could still be characterized as disgusting but a few were almost palatable. By the third generation almost half of the mixtures were decent. Most of these were dominated by one or more strong flavors but in some cases more subtle combinations were starting to be noticeable. Secondary issues like aftertaste also began to play a more important part in the selection process. On the sixth generation the top choice didn't change and the differences between the top three choices were extremely small. After the seventh generation the top choice once again remained the same and we decided to terminate the project.

This pattern of convergence is partially reflected in the plot of the sum of the absolute differences in Figure 1. The sharp peak at the first generation indicates that a relatively good region of the solution space has been found. The slow, erratic decline after this point, however, masks the important work that is being done to improve the taste of the top choice.

Figure 1. The sum of the absolute differences between the current top choice and the previous top choice.



Because of the subjective nature of the fitness function in this project and the huge number of possible mixtures, 15^{49}, most traditional methods of optimization can not be used. Evolutionary algorithms provide a simple, quickly convergent alternative. Although these techniques have previously been applied to art and music, we believe that this is the first time that evolutionary programming has utilized the sense of taste. The rapid change from unpalatable swill to delicious beverage certainly makes this work an interesting demonstration of evolutionary algorithms.

REFERENCES

[1] Fogel, Lawrence J., Owens, Alvin J., and Walsh, Michael J., Artificial Intelligence Through Simulated Evolution, Wiley, 1966.

[2] Dawkins, Richard. The Blind Watchmaker. Penguin Books, 1986.

[3] Sims, Karl. “Artificial evolution for computer graphics” ACM Computer Graphics 25(4), 319-328, 1991.

[4] Todd, Stephen and Latham, William. Evolutionary Art and Computers. Academic Press, 1992.

[5] Graf, Janine and Banzhaf, Wolfgang. “Interactive evolution of images”, in Proceedings of the Fourth Annual Conference on Evolutionary Programming, editor D. B. Fogel, 1995.

[6] Tabuada, P., Alves, P., Gomes, J. and Rosa, A. “3D Artificial Art by Genetic Algorithms”, in Proceedings of the Workshop on Evolutionary Design at Artificial Intelligence in Design - AID' 98, 1998.

[7] Coates, Paul. “Using Genetic Programming and L-Systems to Explore 3D Design Worlds” in CAADFutures '97 editor R. Junge, Kluwer Academic, 1997.

[8] Rosenman, M.A. “An Exploration into Evolutionary Models for Non-routine Design” in Evolutionary Algorithms in Engineering Applications, Eds. D. Dasgupta and Z. Michalewicz, Springer-Verlag, 1997.

[9] Bentley, Peter. Evolutionary Design by Computers. Morgan Kaufmann, 1999.
[10] Graham, I.J., Wood, R.L. and Case, K. “Evolutionary Form Design: The Application of Genetic Algorithmic Techniques to Computer Aided Product Design”, in Proceedings of the 15th National Conference on Manufacturing Research (NCMR), Advances in Manufacturing Technology Vol. 13, September, 1999.
[11] Corne, David and Bentley, Peter. Creative Evolutionary Systems. Morgan Kaufmann, 2001.
[12] The entire issue of Contemporary Music Review, 22(3) 2003.