C2.10 Interactive Evolution

Wolfgang Banzhaf

University of Dortmund

Abstract

We present a different approach to directing the evolutionary process through interactive selection of solutions by the human user. First the general context of interactive evolution (IE) is set, then the standard IE algorithm is discussed together with more complicated variants. Finally, several application areas are discussed and uses for the new method are exemplified using design from the literature.

Keywords

Evolutionary computation, Interactive Evolution, Computer Graphics, Design

C2.10.1 Introduction

The basic idea of interactive evolution (IE) is to involve a human user on-line into the variation—selection loop of the evolutionary algorithm (EA). This is to be seen in contrast to the conventional participation of the user prior to running the EA by defining a suitable *representation* of the problem, C1 the *fitness criterion* for evaluation of individual solutions and corresponding *operators* to improve B2.1 fitness quality. In the latter case, the user's role is restricted to passive observation during the EA C3 run.

The minimum requirement for interactive evolution is the definition of a problem representation, together with a determination of *population parameters* only. *Search operators* of arbitrary kind as E1 well as *selection* according to arbitrary criteria might be applied to the representation by the user. C3 The process is much more comparable to the creation of a piece of art, e.g., a painting, than to the C2 automatic evolution of an optimized problem solution. In IE, the user assumes an active role in the search process. At the minimum level, the IE system must hold present solutions together with variants presently generated or considered.

Usually, however, automatic means of variation, i.e., evolutionary search operators using random events, are provided with an IE system. In the present context we shall require the existence of automatic means of variation by operators for *mutation* and *recombination* of solutions which are C3.2 to be defined prior to running the EA. C3.3

C2.10.2 History and prospects

Dawkins (1986) was the first to consider an elaborate IE system. The evolution of *biomorphs*, as he called them, by IE in a system that he had originally intended to be useful for the design of tree-like graphical forms has served as a prototype for many systems developed subsequently. Starting with the contributions of Sims (1991) and the book of Todd and Latham (1992), computer art developed into the present major *application area* of IE. F1

Later, interactive evolution of grammar-based structures has been considered (Nguyen et al. 1994 and McCormack 1994). Raw image data have been used more recently for the purpose of evolving forms (Graf and Banzhaf 1995). It is anticipated that IE systems for the purpose of engineering design will be moving into application in the second half of the 1990s.

C2.10.3 The problem

The problem IE is trying to address has been encountered in all varieties of evolutionary algorithms that make use of automatic evolution: the existence of non-explicit conditions, i.e., conditions that are not formalizable.

- The absence of a human user in steering and controlling the process of evolution sometimes leads to unnecessary detours from the goal of global optimization. In most of these cases, human intervention into the search and selection process would advance the search rather quickly and allow faster *convergence* onto the most promising regions of the *fitness landscape*, B2.4 or, sometimes, leaving a local optimum. Hence, a mobilization of human knowledge can be B2.7 achieved by allowing the user to participate in the process.
- Many design processes require subjective judgement relying on human intuition, aesthetical values or taste. In such cases, the fitness criterion cannot be formulated explicitly, but can only be applied on a comparative case-by-case basis. Direct human participation in IE allows for machine-supported evolution of designs that would otherwise be completely manual.

Thus, IE can be used to (i) accelerate evolutionary algorithms and (ii) in some areas to enable application of evolutionary algorithms altogether.

C2.10.4 The IE approach

Selection in a standard interactive evolution system, as opposed to that in an automatic evolution system, is based on user action. It is typically the selection step that is subjugated to human action, although in less frequent cases also the variation process might be done by hand.

The standard algorithm for IE reads (following the notation in the *Introduction*): B1.1

Input: $\mu, \lambda, \Theta_\iota, \Theta_m, \Theta_r, \Theta_s$
Output: $\vec{a}^*$, the individual last selected during the run, or
P^*, the population last selected during the run.

$t \leftarrow 0$;
$P(t) \leftarrow \text{initialize}(\mu)$;
while $(\iota(P(t), \Theta_\iota) \neq \text{true})$ **do**
  Input: Θ'_r, Θ'_m
  $P'(t) \leftarrow \text{recombine}(P(t), \Theta_r, \Theta'_r)$;
  $P''(t) \leftarrow \text{mutate}(P'(t), \Theta_m, \Theta'_m)$;
  Output: $P''(t)$
  Input: Θ'_s
  $P(t+1) \leftarrow \text{select}(P''(t), \mu, \Theta_s, \Theta'_s)$;
  $t \leftarrow t + 1$;
od

As in an automatic evolution system, there are parameters that are required to be fixed a priori: $\mu, \lambda, \Theta_\iota, \Theta_m, \Theta_r, \Theta_s$. There are, however, also parameters subject to change $\Theta'_m, \Theta'_r, \Theta'_s$ depending on the user interaction with the IE system. Both parameter sets together determine the actual effect of mutation, recombination, and selection operators.

A simple variation of the standard algorithm shown above is to allow for population parameters to be also the subject of user interaction with the system. For example, some systems (Graf and Banzhaf 1995) consider growing populations and a variable number of variants.

A more complicated variant of the standard algorithm would add a sorting process of variants according to a predefined fitness criterion. One step further is to allow this sorting process to result in a pre-selection in order to present a smaller number of variants for the interactive selection step. Both methods help the user to concentrate his or her selective action on the most promising variants according to this predefined criterion.

This algorithm is formulated as follows:

Input: $\mu, \lambda, \eta, \Theta_\iota, \Theta_m, \Theta_o, \Theta_r, \Theta_s$
Output: $\vec{a}^*$, the individual last selected during the run, or
P^*, the population last selected during the run.

$t \leftarrow 0;$
$P(t) \leftarrow \text{initialize}(\mu);$
while $(\iota(P(t), \Theta_\iota) \neq \text{true})$ **do**
  Input: Θ'_r, Θ'_m
  $P'(t) \leftarrow \text{recombine}(P(t), \Theta_r, \Theta'_r);$
  $P''(t) \leftarrow \text{mutate}(P'(t), \Theta_m, \Theta'_m);$
  $\vec{F}(t) \leftarrow \text{evaluate}(P''(t), \lambda);$
  $P'''(t) \leftarrow \text{sort}(P''(t), \Theta_o);$
  $P''''(t) \leftarrow \text{select}(P'''(t), \vec{F}(t), \mu, \eta, \Theta_s);$
  Output: $P''''(t)$
  Input: Θ'_s
  $P(t+1) \leftarrow \text{select}(P''''(t), \mu, \Theta_s, \Theta'_s);$
  $t \leftarrow t + 1;$
od

The newly added parameter Θ_o is used here to specify the predefined order of the result after evaluation according to the predefined criterion. As before, the Θ'_x-parameters are used to specify the user interaction with the system. η is the parameter stating how many of the automatically generated and ordered variants are to be presented to the user. If $\mu + \lambda = \eta$ in a $(\mu + \lambda)$-*strategy*, B1.3
or $\lambda = \eta$ in a (μ, λ)-strategy all variants will be presented for interactive selection. If, however, $\mu + \lambda > \eta$ and $\lambda > \eta$ respectively, solutions would be preselected and we speak of a hybrid evolution system (having elements of automatic as well as interactive evolution). Other parameters are used in the same way as in the standard algorithm.

C2.10.5 Difficulties

The second, more complicated version of interactive evolution requires a predefined fitness criterion, in addition to user action. This trades one advantage of IE systems for another: the absence of any requirement to quantify fitness for a small number of variants to be evaluated interactively by the user.

Interactive systems have one serious difficulty, especially in connection with the automatic means of variation that are usually provided: whereas the generation of variants does not necessarily require human intervention, selection of variants does call the attention of the user. Due to psychological constraints, however, humans can normally select only from a small set of choices. IE systems are thus constrained to present only at the order of 10 choices at each point in time from which to choose. Also sequentially, only a limited number of generations can be practically inspected by a user before becoming tired.

It is emphasized that this limitation must not mean that the generation of variants has to be restricted to small numbers. Rather the variants have to be properly ordered at least, for a presentation of a subset that can be handled interactively.

C2.10.6 Application areas

An application of interactive evolution may be roughly divided into two parts:

(i) Structural evolution by discrete combination of predefined elements.
(ii) Parametric evolution of genes coding for quantifiable features of the phenotype.

All application use these parts to various degrees.

In the first part, one has to define the structure elements that might be combined into a correct genotype. Examples are symbolic expressions coding for appearance of points in an image plane (Sims 1991) or elementary geometric figures like cone and cube (Todd and Latham 1992). In the second part, parameters have to be used to further specify features of these structural elements. Together, this information constitutes the genotype of the future design hopefully to be selected by a user. In a process called *expression* this genotype is then transformed into an image or three-dimensional form that can be displayed as a phenotype for the selection step.

Table C2.10.1 gives an overview of the presently used interactive evolution systems. The reader is advised to consult details with the sources given in the reference list.

Table C2.10.1. Overview of different IE systems

Application	Genotypic elements	Phenotype	Source
Life-like structures	line drawing parameters	Biomorphs	Dawkins 1986
Textures, images	math. functions, image processing op.	(x,y,z) pixel values	Sims 1991
Animation	math. functions, image processing op.	(x,y,z) pixel values	Sims 1991
Person tracking	(position of) facial parts	face images	Caldwell and Johnston 1991
Images, sculptures	geometric forms and visually defined graphical elements	3D rendering of grown objects	Todd and Latham 1992
Dynamical systems	CA rules, differential equations	system behaviour	Sims 1992
Images, animation	Rules, parameters of L-systems	rendered objects	McCormack 1994
Airplane design	structural elements,e.g. wings, body	airplane drawings	Nguyen and Huang 1994
Images, design	Tiepoints of bitmap images	Bitmap images	Graf and Banzhaf 1995

Figure C2.10.1 illustrates some results with runs in different IE systems.

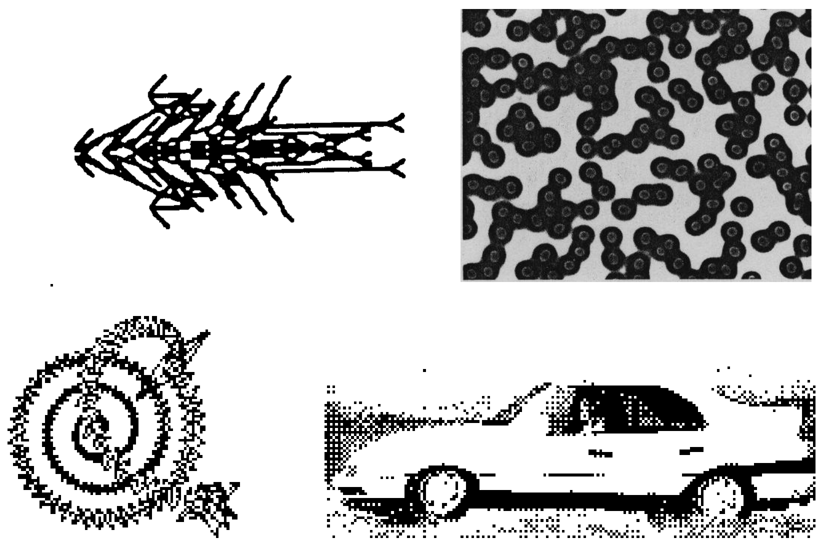

Figure C2.10.1. Samples of evolved objects: (a) Evolved Biomorph (Dawkins 1986); (b) Dynamical System, Cell structure (Sims 1992); (c) Artwork by Mutator (Todd Latham 1992); (d) Hybrid car model (Graf Banzhaf 1995a).

Within the process of genotype-phenotype mapping a (recursive) developmental process is sometimes applied (Dawkins 1986, Todd and Latham 1992) whose results are finally displayed as the image for selection.

C2.10.7 Further developments and perspectives

As of now, the means to generate a small group of variants from which to choose interactively are still not very good. For example, one could imagine a tool for categorizing variants into a number of families of similar design and then present only one representative from each family. In this way, a large population of variants could be used in the background which is invisible to the user but might have beneficial effects in the course of evolution.

Another very interesting area of research is to assign a posteriori effective fitness values to members of the population, depending on user action. An individual which is selected more often would be assigned a higher fitness than an individual which is not. This might result in at least a crude measure of the non-quantifiable fitness measures that lie at the heart of interactive evolution. One might even adjust the effect the operators have on the population, based on what is observed in the course of evolution directed by the user. In this way, an "intelligent" system could be created, that is able to learn from actions of the user how to vary the population in order to arrive at good designs.

Another direction of research is to look into involving the user not (only) into the selection process, but into the variation process. Quite often, humans would have intuitive ideas for improvement of solutions when observing an automatic evolutionary process taking its steps. These ideas might be used to cut short the search routes an automatic algorithm is following. For this purpose, a user might be allowed to intervene into the process at appropriate interrupt times.

Finally, all sensory inputs could be used for interactive evolution. The systematic variation of components of a chemical compound that specifies an odor, for example, could be used to evolve a nice smelling parfume. Taste could as well be subject to interactive evolutionary tools, as could other objects if appropriately mapped to our senses (for instance by virtual reality tools).

With the advent of interactive media in the consumer market, production-on-demand systems might one day include an interactive evolutionary design device that allows the user not only to customize a product design before it goes into production, but also to generate his or her own original design that has never been realized before and usually will never be produced again. This would open up the possibility of evolutionary product design by companies which track their customers' activities and then distribute the best designs they discover.

References

Caldwell C and Johnston V S 1994 Tracking a Criminal Suspect through "Face-Space" with a Genetic Algorithm *Proc. 4th International Conference on Genetic Algorithms, San Diego* R K Belew and L B Booker, Eds. (San Mateo: Morgan Kaufmann) 416–421

Dawkins R 1986 *The Blind Watchmaker* (New York: Norton)

Graf J and Banzhaf W 1995 Interactive Evolution of Images *Proc. 4th Conference on Evolutionary Programming, San Diego* J R McDonnell, R G Reynolds and D Fogel, Eds. (Cambridge: MIT Press) 53–65

Graf J and Banzhaf W 1995a An Expansion Operator for Interactive Evolution *IEEE Int. Conf. on Evolutionary Computation, Perth/Australia* (Piscataway: IEEE Press) 798–802

McCormack J 1994 Interactive Evolution of L-System Grammars for Computer Graphics Modelling *Complex Systems* T Bossomaier and D Green, Eds. (Singapore: World Scientific) 118–130

Nguyen T C and Huang T S 1994 Evolvable 3D Modeling for Model-Based Object Recognition Systems *Advances in Genetic Programming* K Kinnear, Ed. (Cambridge: MIT Press) 459–475

Sims K 1991 Artificial Evolution for Computer Graphics *Computer Graphics* **25**(4) 319–328

Sims K 1992 Interactive Evolution of Dynamical Systems *Toward a Practice of Autonomous Systems* F J Varela and P Bourgine, Eds (Cambridge, MA: MIT Press) 171–178

Todd S and Latham W 1992 *Evolutionary Art and Computers* (London: Academic Press)

Further reading

This section is intended to give an overview of presently available work in interactive evolution and modelling methods which might be interesting to use.

1. Prusinkiewicz P and Lindenmayer A 1991 *The Algorithmic Beauty of Plants* (Berlin: Springer)

   An informative introduction to L-systems and their use in computer graphics.

2. Koza J R 1992 *Genetic Programming* (Cambridge, MA: MIT Press)

   A book describing methods to evolve computer code, mainly in the form of LISP type S-expressions.

3. Caldwell C and Johnston V 1991 Tracking a Criminal Suspect through face-space with a Genetic Algorithm *Proc. International Conference on Genetic Algorithms 1991 San Diego* R K Belew and L B Booker, Eds. (San Mateo: Morgan Kaufmann) 416–421

   Very interesting work containing one of the more profane applications of interactive evolution.

4. Baker E 1993 Evolving Line Drawings *Proc. International Conference on Genetic Algorithms 1993 Urbana-Champaign* Forrest S, Ed. (San Mateo: Morgan Kaufmann) 627

   This contribution is discussing new ideas on design using simple style elements for interactive evolution.

5. Roston G P 1994 *A Genetic Methodology for Configuration Design* Doctoral Disseration, Carnegie-Mellon University, Pittsburgh PA, 1994

   Informed discussion of different aspects of using Genetic Algorithms for design purposes.

