Interactive Evolutionary Computation : a survey of existing theory

Yann SEMET

November 13th, 2002

Abstract

A literature survey of non applicative research is conducted on all attempts that have been made to enhance the design of Interactive Evolutionary Algorithms (IEAs) by theoretical means. Emphasis is put on theory and builds over the exhaustive application oriented survey made by Takagi [36]. After having positioned the study in its background and described the only attempt of mathematical modelling that was made [28], previous work is inventoried in the three main, up to now explored, directions of research : enhancing the system's interface, allowing the user to actively participate in the search and boosting the search itself.

1 Introduction

For their robustness and ability to tackle huge, fuzzy and highly non-linear search spaces, Evolutionary Algorithms (EAs) are especially adapted to solve optimization problems in the field of design where excellence is measured in terms of subjective evaluation of the solution by the user. The problem is that the introduction of a human component as a part of an iterative computation raises several issues such as inconsistence, slowness and, more importantly , limitation of energy (computers are tireless, human beings are not).

There are two main ways to allow a human operator to interact with a GA. The first one, the most straightforward and commonly used, consists in having the user assign fitness to individuals. The human operator is in this case only here to provide his subjectivity and to evaluate the quality of the solutions found so far (the individuals in the population) in order to provide the GA with relevant information to pursue its selecto-recombinative search. This framework is known as *narrow definition of IEC* and can go from simple human selection (the operator simply points at the best individuals) to exhaustive evaluation of individuals on a fine scale (e.g. from 1 to 100). It has also been proven useful in a certain number of studies that the user can be more active by helping the search itself. In that case, also known as *broad definition of IEC*, the user has more control on the behaviour of the search and can influence it, for example by freezing the evolution of some genes.

The first problem one might think of that would be yielded by having a human based fitness function is that given the fuzzy and unstable character of human subjective appreciation, the corresponding fitness landscape hosting the search would be accordingly noisy and unpredictable. This issue is not however to be taken too seriously as it is largely compensated by the fact that human beings are all the less exigent than their judgement is fuzzy. The user is not actually looking for a precisely defined numerical target but rather for a subset of the search space that gives the general impression he is looking for. The fuzzy aspect of human subjectivity is therefore more to be taken as a basis for robustness than as a source of trouble.

There is however a major problem with IEC which is that humans' judging abilities are strongly limited both in time and space. This results in extremely stringent limitations on population size (typically not more than 10-20) and on the allowable number of generations (typically 20-25, after what users get bored and tired and their appreciation of solutions becomes consequently fuzzier and less coherent). This issue, also known as *human fatigue* is the key-obstacle to the design of competent interactive EAs.

In spite however of these immediate obstacles constituted by human fuzziness and fatigue, interactive have been widely and successfully applied to number of real-world cases, the main categories of which are outlined in the next section.

2 Background and applications

Several main trends have guided the development of IEC techniques in the last couple of decades. We will try here to give the main directions of research where IEAs have been successfully applied and promise to do so again in the future. This section does not however aim at conducting an extensive survey of application-oriented research, one can refer to [36] for this purpose, but rather at giving the reader a general idea of the reason why these algorithms are worth being understood and enhanced. Besides being this necessary background, this overview is also meant to outline the main problems that are recurrently encountered by designers of such systems in order to give a motivation for the theoretical research that is going to be described next.

If Genetic Algorithms are introduced in the mid 60's, it's only in the mid 80's that people start to think of Evolutionary Algorithms as an interesting tool for interactive design systems, event in which the publication in 1986 of "The Blind Watchmaker" by Richard Dawkins ([6]) seems to have played some kind of a role. This notorious book, built as an argument against creationism, reveals the abilities of natural evolution as a slow but incredibly efficient and imaginative designer. Dawkins also introduced, to illustrate his point, the famous Biomorphs : first virtual creatures with incredible shapes to be evolved by an Evolutionary Algorithm (a simple selecto-mutative GA). Little work was done afterwards until the mid 90's. Statistics given in [36] indeed indicate that out of 251 inventoried works, only 31 were done prior to 1995. After a peak in 1999 (57 papers published), it seems the production volume in this area is stabilizing around a couple dozens of papers per year.

The development of IEC is historically linked to the field of computer graphics and image processing, which field remains today along with industrial design the main source of development in this area. This spontaneous connection is easily understandable when noticing that images are especially adapted to a search mechanism like that of GAs where exploration is performed by recombining highly fit components. Besides being trivially spatially decomposable (an image is composed as an aggregation of several visually and/or semantically distinct elements like the parts of a face), images and their potential manipulations are easily describable in terms of parameters (wavelet coefficients, contrast levels, convolution filters, etc.), which can be straightforwardly used as genes. An excellent exemple of this is the famous face-montage system designed by

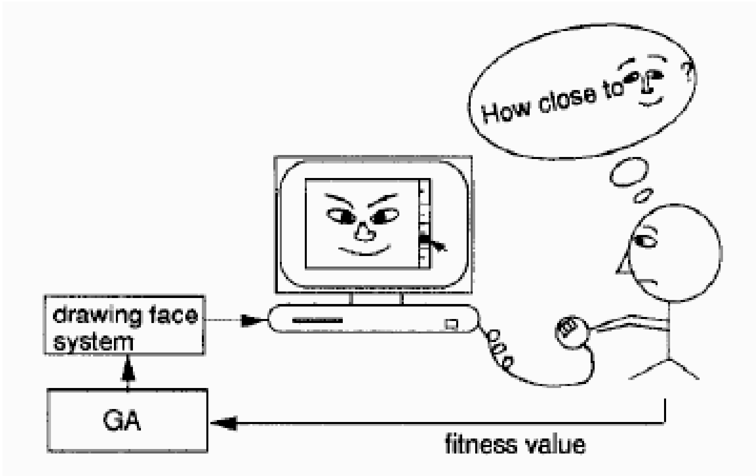

Figure 1: A typical IEC System : the user operates a face montage system to reconstruct a particular face he has in mind.

Caldwell and Johnston ([4]). This work is also very representative of IEC systems in general and can be kept in mind to illustrate the forthcoming theoretical discussion. A similar system is besides actually used as an example application in several of the papers that are presented next. This system is illustrated by figure 1. The user wants to retrieve a particular face he has in mind using a GA to browse a database of face parts (eyes, chins, etc.) and to evolve a small population of candidate faces. Another reason why image (or visual design in general) related problems are so well suited for IEC in general is that evolving an image yields direct visual feedback of the solutions's quality making fitness evaluation straightforward and even almost parallel when several individuals are displayed next to each other on the screen.

One can't really talk about Interactive EC and its application to computer graphics without mentioning the work of one of its most notorious artists, Karl Sims who did a great job showing EAs were great tools for assisted graphical creativity ([33]).

Beyond visual design, industrial and engineering design in general is a great application field for interactive EC (see [24]) and by itself provides a pragmatical justification for research topics related to competent IEC.

Another wide field for applications, with strong commercial potential opportunities, is that of database retrieval, which fits, in a more general way, within the framework of data mining. In this particular kind of applications, the strength of genetic search is used along with its ability to quickly browse huge, heterogeneous search spaces.

A word finally, should be said to underline the fact that the research of a competent IEC is independent of the particular paradigm that is used, may it be GAs, ESs or GP. They all share the same global dynamics and have all been successfully applied on several distinct cases (see for example [4, 6] for GAs, [12, 13] for ESs and [27, 39, 23] for GP). An exception, however, should be made for the particular case of Ant Colonies that first of all, although still pertaining to the field of Evolutionary Computation, don't have the same mechanics, and second of all , probably for being a little bit too young of a field, haven't been yet applied to interactive systems (a few examples although start to appear, see for example [31, 30]).

IEC is a strongly application oriented research topic, and theory should not be derived without relying on a strong experimental background. A paper by O'Reilly and Bentley ([1]) provides a great transition in this regard before we move on to theory. They provide a set of 10 guidelines, drawn out their experience in using GP as a form design strategy ([23]) (among other things) and enumerate the precautions that should be taken in any case for the successful setup and use of an evolutionary design system, for example giving hints on how to set an appropriate balance of control over the results between the user and the GA.

Now that the problem is posed, we can proceed to the next section where a more rigourous mathematical journey is undertaken.

3 Mathematical framework

3.1 Mealy Automata

The first step of competent practice has to do with mathematical formalization. Having a good model indeed allows to predict the behaviour of the GA, to prove its ability to converge, to derive sizing equations and so on. Several tools are available to the EC practitioner to achieve such mathematical descriptions : difference equations (e.g., [8], [9], [3]) , Markov chains, statistical mechanics ([26]), etc. Markov chains in particular are one of the most commonly used framework to completely describe (see [16] for mathematical fundations, [10] for a good first example and [5] for a more fancy and recent one), through sets of state-transition matrices, the run of the GA (see also [21]). It's although natural to ask oneself whether this tool is well suited for the particular case of Interactive EC. This issue was addressed in [28] using as we will see, something slightly more general than Markov Chains : Mealy Automata.
Markov Chains belong to the general class of stochastic automata (a more rigorous introduction can be found in [25]). A stochastic automata can be described as a tuple

$$< S, X, Y, P\{s', y|s, x\} >$$

where S is a state space, X a set of input symbols, Y a set of output symbols and where P represents the probabilities that the system goes from state s with input x to state s' with output y. Markov chains correspond to the particular case where X and Y are empty (the process is then fully determined by itself) and Mealy Automata to the particular case where transition probabilities to state s' and output symbol y are mutually independent, which can be expressed by :

$$P\{s', y|s, x\} = P\{s'|s, x\}.P\{y|s, x\}$$

with

$$P\{s'|s, x\} = \sum_{y \in Y} P\{s', y|s, x\}$$

and

$$P\{y|s, x\} = \sum_{s' \in S} P\{s', y|s, x\}$$

.
Having no way of representing interaction with the user through inputs and outputs since the corresponding X and Y sets are empty, simple Markov Chains

are evidently insufficient to fully characterize Interactive Evolutionary Computation, while it's obvious that this interaction can be modelled in the general framework of stochastic automata.

3.1.1 Abstract model

The stochastic automata used as model for IEC in [28] can be fully described by giving the five elements of its representative tuple. The state space S simply represents all the configurations the population can take. The output space Y has no particular significance here and can be left as empty. The input set X is where the human being intervenes : the model developed in [28] has it describe the selection that is made over the population : to each individual corresponds a number that tells how many exemplars of this individual are going to be used for the mating phase. The transition probabilities, finally, can be represented by a set of transition matrices $A(x) = U(x).M$ where x is the input, where U stands for the selection operation and M for the remainder of the GA (crossover and mutation). The latter matrix is known, having been previously derived in the Markov Chains/GAs literature (see [21]).
The output set being empty, our automaton belongs to the set of stochastic Mealy automata, which allows to conveniently model the successive interventions of the user (through selection performed at every generation) using the fact that $A(vw) = A(v).A(w)$ where A are transition matrices (see upper) and v and w successive inputs by the user. This property, which can be seen as the equivalent of the Chapman-Kolmogorov equation from the Markov Chains framework allows to easily describe the movement of the Interactive GA through its search space as driven by a human operator.

3.1.2 Explicit model

To illustrate the abstract model, an example of transition matrices derivation is given for a population of size 2 composed of boolean individuals (boolean strings of length 1). Using the traditional derivation, calculating in particular the probability of obtaining the zero string, the transition matrix for crossover and mutation is obtained, with mutation probability μ per bit and a uniform crossover (with probability $\xi = 1/2$ of choosing either parent) :

$$M = \begin{array}{c} \\ 00 \\ 01 \\ 10 \\ 11 \end{array} \begin{array}{c} \begin{array}{cccc} 00 & 01 & 10 & 11 \end{array} \\ \begin{pmatrix} (1-\mu)^2 & \mu(1-\mu) & \mu(1-\mu) & \mu^2 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ \mu^2 & \mu(1-\mu) & \mu(1-\mu) & (1-\mu)^2 \end{pmatrix} \end{array}$$

As an example, here is the selection matrix for the case where the user picks the first individual twice :

$$U(2,0) = \begin{array}{c} \\ 00 \\ 01 \\ 10 \\ 11 \end{array} \begin{array}{c} \begin{array}{cccc} 00 & 01 & 10 & 11 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

which yields, for the final transition matrix :

$$A(2,0) = U(2,0).M = \begin{array}{c c} & \begin{array}{cccc} 00 & 01 & 10 & 11 \end{array} \\ \begin{array}{c} 00 \\ 01 \\ 10 \\ 11 \end{array} & \begin{pmatrix} (1-\mu)^2 & \mu(1-\mu) & \mu(1-\mu) & \mu^2 \\ (1-\mu)^2 & \mu(1-\mu) & \mu(1-\mu) & \mu^2 \\ \mu^2 & \mu(1-\mu) & \mu(1-\mu) & (1-\mu)^2 \\ \mu^2 & \mu(1-\mu) & \mu(1-\mu) & (1-\mu)^2 \end{pmatrix} \end{array}$$

By deriving such matrices for every possible input (i.e. (1,1) and (0,2)), one can therefore obtain a full description of what can happen with this elementary Interactive GA.

3.1.3 Contribution

The contribution of this work is, to the confession of the author himself, rather unclear. Two suggestions are however made. First of all, the stochastic automata literature teaches us that there exist techniques that allow to decompose such an automaton into a controlled random source and a regular deterministic automaton, which might allow to implement proper realizations of Interactives EAs and to get information and control about the random side of those systems. Secondly, convergence proofs and conditions could be derived from such an explicit model to help an ensuring a proper design of the underlying EA of IEC systems.
But more generally, this work has the merit of being the first and only attempt to rigorously describe IEAs and as such deserves to be carefully considered as a potential source of ideas for more theoretical work in the field of IEC.

4 Enhancing the User Interface

The purpose of this section is to describe the efforts made by researchers to ameliorate the Graphical User Interface (GUI) of IEC systems in order to make it easier for the user to browse through individuals and generations. And if every IEC systems calls its own effort to design a convenient GUI, two works are described here that are general enough to be applied in pretty much any case.

4.1 Fitness granularity

The classical IEC system consists in having the user input fitness values for each and every individual in the population and it is consequently reasonable, as a competent practitioner to put some extra thought on how these values are input. The work conducted in [38] questions the degree of continuity these values shall have. It is important to understand that it is a psychological acception of the concept of continuity we are dealing here with. The values are quantized in any case but a distinction is made between systems where the different categories (typically 4 or 5 like : "very bad", "bad", "good" and "very good") are clearly separated and systems where close categories aren't really distinguishable from each other (typically scales from 0 to 100). The former case is called "discrete fitness evaluation", the latter "continuous fitness evaluation".
Two methods are proposed in [38] : the first ones allows discrete fitness input

and is compared to the traditional one with continuous input, the idea being to simplify the user's task so that he doesn't have to bother with quantifying slight differences and therefore can perform more evaluations, which means more generations for the GA to ensure proper convergence. The problem is that noise results from this thick fitness measure and that convergence is consequently disturbed : signal is weakened, causing the GA to wander more blindly and elitist strategies are impaired when the population is homogeneous because elites simply cannot be detected.
The second method take these remarks into account and uses both discrete and continuous input. The former is useful in early generations when evaluation can remain coarse without slowing down convergence and is later replaced, at the user's convenience, by a more refined evaluation scale.
Subjective tests are conducted that verify that the second method is significantly better, in terms of convenience, than pure methods (i.e. methods using only one kind of input).

4.2 Fitness prediction

An original idea is proposed in [22] to improve the comfort of the search for the human operator who is supposed to provide fitness evaluation. Noticing that the order in which the individuals are presented to the user can have a strong influence on her/his feeling of comfort and therefore on her/his ability to perform more evaluations, Ohsaki and Takagi suggested to try to roughly predict the fitness which is going to be given to each individual in order to present these individuals in approximately right order for the actual process of evaluation, guessing it would make it way easier and faster. Figure 2 illustrates this idea.

4.2.1 Two methods of prediction

Two ways to proceed are proposed in [22] to achieve the prediction phase. The first one consists in training an Artificial Neural Network using individuals drawn from previous generations. When the GA produces new individuals, those are fed to the Neural Network (a multilayer perceptron) which predicts their future fitness. Once those fitnesses are actually evaluated, back propagation is applied to the Network's weights. Experimental results show that this technique works best with 10 hidden neurons and with individuals taken only from the previous generation (k-1). The latter constatation makes sense as the current local search space is closer to that of recent generations than to that of earlier ones.
The second method of prediction uses elementary interpolation. Each individual's fitness is estimated using an average of its neighbor's fitnesses weighted by euclidean distance (the closer is an individual to another, the likelier it is they are going to have a similar fitness).
Experiments are conducted with a simulation test without human intervention. Results compare three orders obtained at each generation : (1) the order in which individuals are generated by the GA (random order fitness-wise) ; (2) the predicted fitness-based order ; (3) the actual fitness-based order. Observation of the correlation factors between those orders (comparison of (1)/(2) and (2)/(3) correlations) show that the two aforementioned methods effectively predict the fitness which is going to be given by the human operator. The second method,

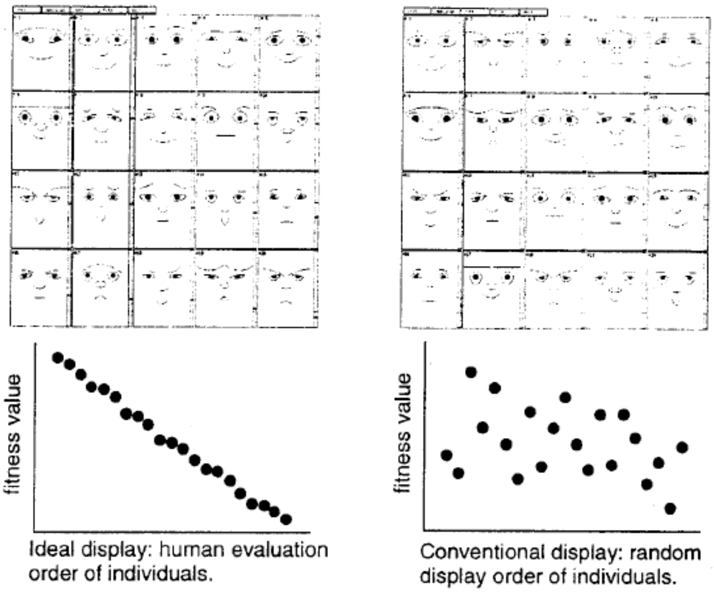

Figure 2: The role of fitness prediction : as illustrated by the underlying graph, faces on the right are presented to the user in random order, as generated by the GA, (a smiling face can be next to an angry one and so on) while faces on the left are sorted according to their predicted fitness values which makes a nice smooth transitions from angry faces to happy ones ; the user then just has to pick the degree he is looking for.

using interpolation based on Euclidean distance is although shown to perform clearly better than the first one, based on Neural Networks.

4.2.2 Subjective Tests

Two subjective tests are then conducted, according to the standards of research in psychology, to assess the ability of the proposed improvement (presentation of individuals to the human operator in the predicted fitness-based order) to improve the quality of the search.
The first test simply replaces the simulation previously used to compute correlation factors by an actual human operator thereby testing the actual efficiency of the prediction method.
The second one draws from comments made by subjects that operated the system and questions the improvement in ease of use. Each subject is asked to evaluate, on a scale from -2 (very difficult) to +2 (very easy), the comfort of use of the two systems (the conventional one, without any modification of the order of presentation, and the proposed one). It has to be underlined that this second test is performed, oddly enough, with a fake Interactive GA : in order to make sure that the only difference between the two experimental settings (conventional and proposed) was the display order, which is close to impossible as human mood and therefore evaluation mechanics varies from one experiment to another, the outputs of Interactive GA (i.e. the individuals suggested by the GA based on the human distribution of fitness) were shortcut to be replaced by the outputs of the first subjective test. The subjects were therefore told they were operating an Interactive system while this system was completely deterministic (even if this deterministic behaviour corresponds to previously recorded human behaviour).

4.2.3 Results of the tests

Unfortunately, results clearly show that there is no statistically significant (according to the sign test) difference : (a) between the "predicted fitness"-based order and the actual "human fitness"-based order ; and (b) between the conventional and proposed methods as regards ease of use.
This works therefore shows a strong and interesting possibility to predict the behaviour of the GA and therefore a potential way to improve the interaction of the system with the human operator but it fails to prove it efficient probably only by lack of a relevant application. Efficience of such a method in the IEC framework is indeed likely to be strongly application dependant.
Several valuable lessons however are drawn by the authors which could lead to extremely interesting improvements of the method. First of all, when trying to understand why the prediction was wrong, they noticed that different importances were accorded to different features during the evaluation and that these importance "weighting" was varying across the run of the Interactive GA (for example the operator would focus on a particular feature at first, giving strong fitnesses to individuals doing well in this feature regardless of other features and then once this feature was optimized enough to his taste across the population, she/he would switch to another feature consequently modifying the importance of the different criteria she/he uses for evaluation). This constatation suggests to use a weighted average instead of the Euclidean distance so that something (the weights) can reflect the relative importance of the different features and the evolution of this distribution of importance with time.
Another conclusion drawn from listening to the remarks made by experimental

subjects is that presenting the individual in a non-random, pre conceived, order might not help and even might entrave the good run of the search. A good example of such a remark is that evaluating an individual situated among dissimilar others (which is the case with the conventional method) might actually be easier as you can rely on local visual contrast.
The observance of such facts, enlighten by this kind of research, could bring a great deal of improvement in IEC practice and also underline what might be the main contribution of this work and which relies in methodology : IEC has to do with the interaction of computers with human beings and as such could greatly benefit from results and methods taken from the field of psychology.

5 Active user intervention

The traditional way to do interactive EC is to have a human operator play the role of the fitness function when the said fitness cannot be calculated by a machine. But there are other ways in which humans can do better than computers and this section gives the example of two IEC systems where the human operator does something else than fitness evaluation and helps the EA to perform a clever search. A synthesis of this research is given in [35].

5.1 Knowledge Embedding

The first idea is developed in [37] and consists in allowing the user to dynamically specify which portion of the search space should be explored at a particular stage of the run. This means that the evaluator is allowed to focus on a particular phenotypic feature of the individuals and to decide to freeze or unfreeze its evolutions. The experiments are conducted on a face montage system similar to the famous one by Caldwell and Johnston ([4]) and at any generation, the user can decide that the eyes or hair is good enough and that it shouldn't evolve anymore (see figure 3). This decision has an impact on the search because it reduces the dimension of the search space thereby yielding faster convergence because the EA is focusing on the important subspace.

Results are encouraging on this particular exemple : four experiments are conducted in [37] to underline it . The first one questions the quality of the search and shows through a subjective test that the proposed method almost always reaches a better solution. The second, also a subjective test, suggests that the system is easier to operate. The third one is a count of the proportion of correct subsolutions across the whole population at different stages of the run and clearly shows that the new system reaches bigger proportions faster. The fourth and final one although, shows that the operation time is significantly higher with the new system, which is obvious as an extra time-taking step is required to freeze the genes adequately but might be compensated by the fact that convergence is faster.
The general idea to be remembered here is that that user can help the search by feeding it with knowledge : not only is the user able to evaluate individuals but he also can, most of the time, say why and tell what features makes an individual good or bad. This knowledge shouldn't be wasted and a great deal of improvement can be brought by formalizing it and incorporating it in the search.
There is a big issue however which lies in the fact that the mapping between phenotype and genotype is not trivial and what is identified as being good or bad is not necessarily trivially encoded by a single gene that can be frozen or unfrozen whenever convenient. This clearly restricts the applicability of online

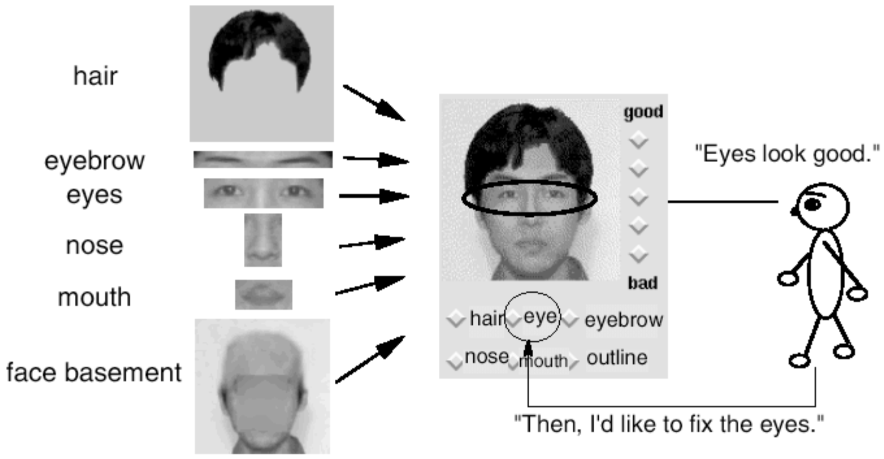

Figure 3: The human operator can decide to freeze whatever feature he thinks has evolved enough. e.g. here the eyes.

knowledge embedding methods to problems with trivial mapping as is the face montage search. This certainly is a call for development of non trivial mapping description methods.

5.2 Visualized IEC

Another thing a human operator can do to help enhancing the search is to use its visualization abilities. By visualizing a proper representation of the search space, the operator could indeed help the GA to avoid local traps and to aim more quickly at interesting search regions. This idea was proposed in [11] and successfully applied to a real-world example (a fitting problem based on geophysical data) in [2].
All the difficulty lies in finding a proper representation for a multidimensional search space, a representation that could be used and understood by the human operator. A possibility is to use two dimensional maps. A great deal of research has been conducted in mapping methods from nD spaces to 2D maps. These methods do not keep all the information contained in the original space of course but try to preserve the topological relationships between individuals so that the output map gives a sufficiently adequate view of the shape of the search space. In [11], Kohonen's Self Organising Maps ([18]) are used. Using the output map, the operator can have an idea of the distribution of fitness values across the search space and point out the right direction of search to the GA. This is achieved in the following way : the operator visually spots a good region in the search space as represented by the Kohonen map, picks an individual there and places it in the population as a new elite, kicking out the worst performing individual. Acceleration of convergence is of course expected as a result of the application of this elitist strategy. The system is illustrated by figure 4.
This idea has been developed with the application to interactive EC in mind but is not restricted to it and could as well be applied to regular EC, i.e. with auto-

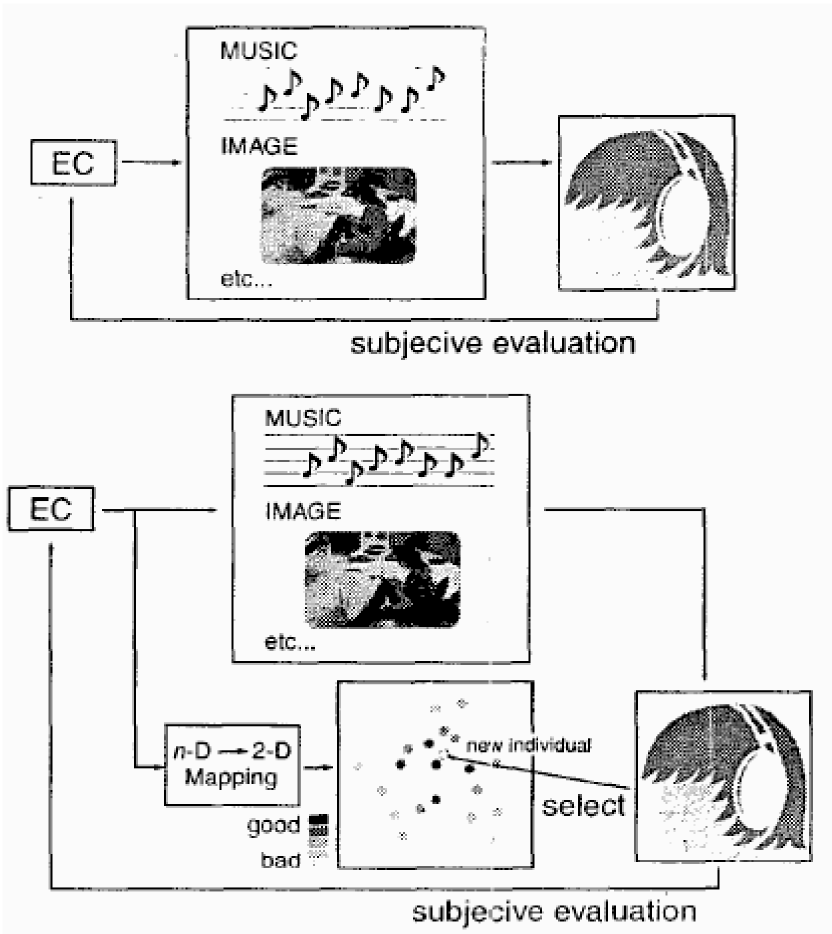

Figure 4: The way visualized IEC works : besides providing fitness evaluation, the human operator visualizes the distribution of fitness across the search space and picks new elites in appropriate regions.

matic fitness evaluation and experiments in [11] are conducted in this framework : the benchmark is not a subjective evaluation but Schaffer's second function. A regular simple GA tries to minimize this function and its performance is compared to that of a "Visualized GA" where the user feeds the population with a visually detected elite at every generation.
Experimental results show that a visualized GA with a population of size 20 consistently performs as well or better than GAs with population sizes of 100 or even of 1000. The different experiments, tend to show that the visualized GA is not impaired when the size of the problem grows while the regular GA is which would mean that the proposed algorithm is especially well suited for complex tasks. These results are fairly encouraging and those obtained in [2], besides being similarly exciting, also suggest that such visualization tools could be useful when dealing with issues such as non-uniqueness. This issue is generally tackled by using expert human knowledge to discard unrealistic solutions and visualization of the search space could provide a great way to embed this knowledge in the search by materializing the topology in terms of different classes of solutions. The human expert can indeed then provide a way to deal with cases where two classes have similar fitness values but when one class is unrealistic or non desirable and by leading the search toward the desired class, the human operator goes further than simply giving out fitness values and thereby enhances the search not only in terms of convergence speed but also in terms of solution quality.

6 Optimizing the search

Interactive evolutionary computation is, before anything else, yet another kind of evolutionary computation and a great deal of research has already been conducted in this area to enhance algorithms, improve their speed or their efficiency. Even if interactive, IEC still has to do with a population of individuals interacting with a fitness landscape and this interaction, as we will see, can be improved at several levels.

6.1 Accelerating convergence by fitting a single peak function

One of the critical issues raised by the introduction of human driven fitness evaluation is, as widely stated before, the little number of generations available to conduct the search of an acceptable solution. It is therefore important to take great care of augmenting the convergence speed of the GA. One trick proposed in [15] consists in accelerating the search using local optimization. It is widely accepted fact that GAs don't go into the real world without hybridization with local search algorithms. In this case, the idea is to approximate the shape of the search space with a quadratic function and to use this approximation and its trivial optimum (it's a single peak function) to feed the population with fresh elite of supposedly highly fit individuals, the aim being to speed up GA convergence, especially in early generations which are crucial to Interactive EC.

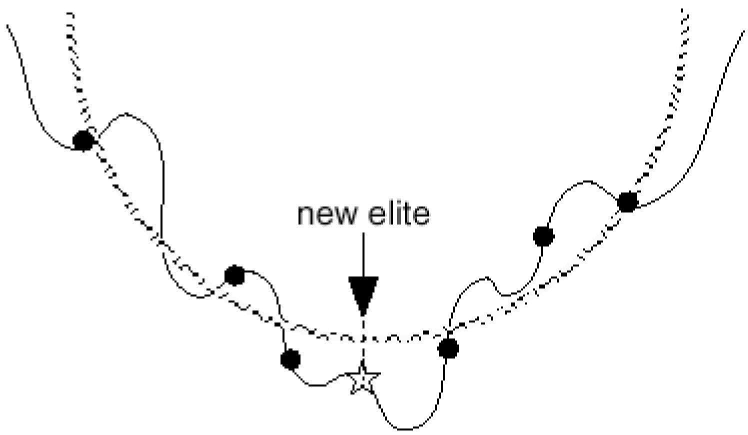

Figure 5: Approximation of the local search space by a quadratic function : a way to pick highly fit new elites.

6.1.1 The method

The idea is to approximate the shape of the search surface as sampled by a given number m of individuals using a polynomial of order 2. n being the number of genes and f the fitness function, the approximation is given by :

$$f(x_1, x_2, ..., x_n) \approx \sum_{i=1}^{n} a_i(x_i - bi)^2 + c \tag{1}$$

A new individual is then obtained by calculating the peak of this approximation (see figure 5). This individual is introduced in the population by replacing the worst performing one, provided of course the latter has an inferior fitness.

6.1.2 Known issues and crucial parameters

This replacement is performed every k-th generation. The parameter k therefore controls the amount of computational effort one wants to spend in local search and thereby allows to control the emphasis that is placed on early generations (provided the value of k can be adapted along the run).
The approximation is computed using m individuals. The parameter m, ranging from a few individuals to the entire current and past population controls the spatio-temporal radius of the local search : if a mimimum amount of information is required, too much of it might lead to irrelevant fitting.
Two methods are tested to calculate the quadratic function's coefficients : Least Square Minimization (LSM) and a simple GA (called subGA as it is integrated to the interactive one). LSM methods demand computation time but provide one single exact solution to the quadratic function's minimization whereas subGA provides several, gradually increasing in quality, approximations whose roughness is not necessarily problematic in the context of Interactive EC.
The final question to be answered as to how the algorithm will proceed is that

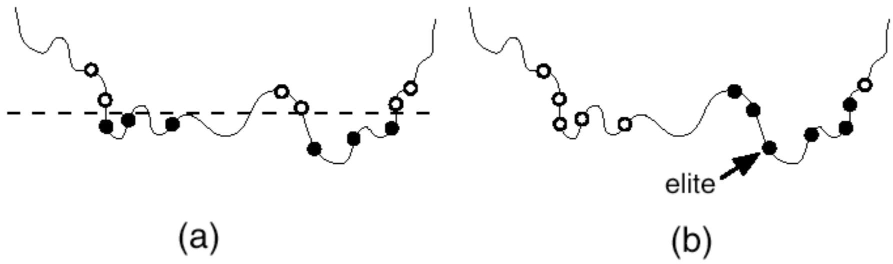

Figure 6: There are two ways to pick the individuals that are going be used to compute the quadratic function : the best-n selection method (a) and the nearest-n selection method (b).

of the selection method for the aforementioned m individuals. Two methods are proposed in [15] : either picking the m best individuals or picking the m individuals that are nearest to an elite in the current generation (nearest-m method) (see figure 6). The latter is an effort to handle cases where there exist a local minimum, close in fitness to the global one but far away from it ; by selecting individuals close to the supposed global optimum, even if their fitnesses are a little inferior on average allows the search to concentrate its full strength on the right spot.

6.1.3 Results and comments

The method, tried on DeJong's five functions ([7]) and on Schaffer's two ([29]) shows consistant and statistically significant improvements on all of the 7 benchmarks.
These encouraging results should be praised for attracting IEC designer's attention on competent GA practice and are a great example of how mathematical care of the underlying mechanics of the GA might bring something, in this case convergence speed, to a human oriented system.
Nevertheless, it is strongly regrettable that those experiments were conducted on this very restricted set of benchmark functions, which are a bit out of date and don't raise issues such as deception or hierarchical traps, and even more regrettable is the absence of tests in an actual IEC framework, i.e. with actual human intervention, all of which strongly reduces the contribution of this potentially relevant work to the particular area of Interactive EC.

6.2 Clustering

Clustering is another interesting and relevant idea drawn from competent GA practice and suggested as applicable to IEC. Parmee et al. mainly, developed Cluster Oriented Genetic Algorithms (COGAs, see for example [24]). Another good example, which we are going to use to illustrate the principle, of this technique was proposed elsewhere in [20] along with a fairly convincing example of image retrieval based on fuzzy human description (e.g. "I'm looking for a gloomy looking picture") and later formalized in [17]. As we saw, one of the critical issue of Interactive EC is the small amount of fitness evaluations a human being is able to perform before he gets tired while Genetic Algorithms

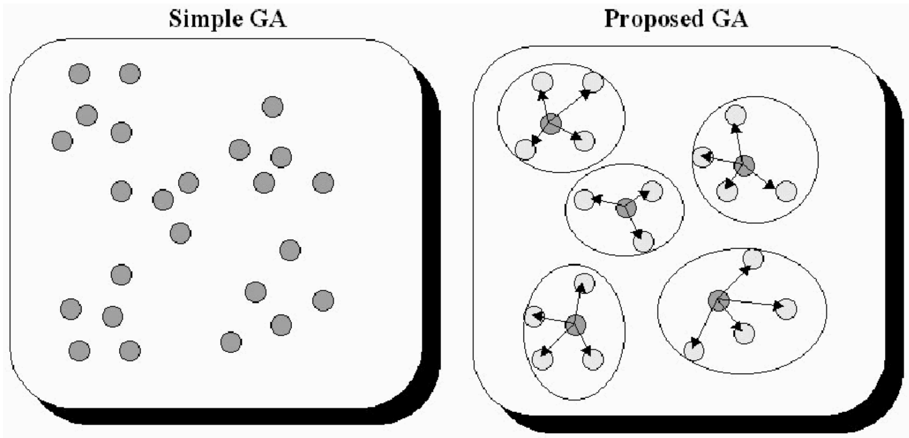

Figure 7: Only one human fitness evaluation per cluster. Remaining fitnesses are computed according to the Euclidean distance to the cluster's representative.

require a huge number of trials and generations to reach acceptable solutions. Another characteristic of human-driven search is its fuzziness and the fact that individuals that are close to each other in the genotype space might not be distinguishable from each other in the phenotype space by human perception. It is with these two facts in mind that the work described in [17] was conducted : the idea is to divide the population into clusters and to perform evaluation on only a few individuals per cluster (see figure 7).

6.2.1 The method

Clustering algorithms consist in grouping samples of a data set so that the samples are similar within each cluster. The particular subclass of clustering algorithms used here is named "Partitional clustering". In this particular class, the algorithm's goal is to divide the data set in several non overlapping partitions. The particular algorithm used in [17] is a simple "k-means clustering". Without giving too many details, it consists in iteratively attributing each sample to a cluster whose centroid is the closest and by recalculating the centroids' positions at each iteration according to the samples that have been related to it. This algorithm therefore provides a way to partition our genetic population into a predefined number of clusters each of those being represented by an average individual (the resulting centroid). Similarity measurement is based on euclidean distance between genotypes. The run of the traditional GA is affected only by the fact that fitness is actually calculated only for each cluster's centroid. In the case of IEC, for example, the user only has to give her/his opinion once per cluster. The remaining individuals are then evaluated after their representative and proportionally to the distance to this representative.

6.2.2 Results

Tests were conducted on De Jong, Griewangk, Rastigrin and Schwefel's traditional benchmarks comparing results obtained by two regular GAs (population size being respectively 100 and 10) and by the proposed hybrid using a population of size 100 but with 10 clusters, which means only 10 fitness evaluations per generation.
Experimental results show that the proposed algorithm systematically behaves as well as the bigger GA (n=100), in particular avoiding local minima where the smaller GA gets trapped (as is the case for example in De Jong's second function).
This work therefore proves the possibility to perform good search with less evaluations which is a potential gold mine for Interactive EC but again, tests are conducted on somewhat rusty benchmarks and no evidence is given as to whether clustering is relevant to search in subjective fitness spaces where topological proximity does not imply subjective proximity. A problem somehow tackled by the research presented in the next section

6.3 Psychometrical spaces

The following works take into account the fact that the way an individual is described by its genotype (a set of parameters) is not necessarily the way it is perceived by a human being. Most of the time there is a non trivial mapping between the two spaces, which causes problem for convergence as the GA follows the human operator in her/his psychometrical space while it is manipulating individuals in the parameter space. This can yield disruptive behaviour of the GA or difficulty for the user to see her/his impressions reflected in the run of the system. The following works ([34] and [37]) try to address this issue by constructing a mapping between the two spaces so that the IEA can get closer to what the user has in mind.

6.3.1 Psychometrical space and fuzzy fitness assignment

The first paper ([34]) aims at reducing the user's burden by automatizing part of the fitness assignment. There are two ways in which a human operator can indicate her/his tastes to the GA : either by rating every single individual or by simply selecting a few among the best ones. This work is in the latter case and aims at automatically this simple selection into a completely automatic fitness assignment over the whole population. This is achieved by spreading the fitness (set by default to 1.0) of the selected individuals to their neighbours according to a set of fuzzy rules (e.g. : if (distance is small AND generation is small) then fitness is 0.5) based on a metric that defines closeness relationships between individuals.
The problem is that to be efficient, this metric has to be defined in terms of subjective closeness ; it has to describe the topology of the psychometrical space, not that of the parameter space. Two individuals living close by in the latter space can indeed be perceived very differently and inversely, two individuals perceived as similar can have a very different set of describing parameters.
The idea is therefore to construct a mapping function :

$$S = g(F)$$

where

$$F = (f_1, f_2, ..., f_n)$$

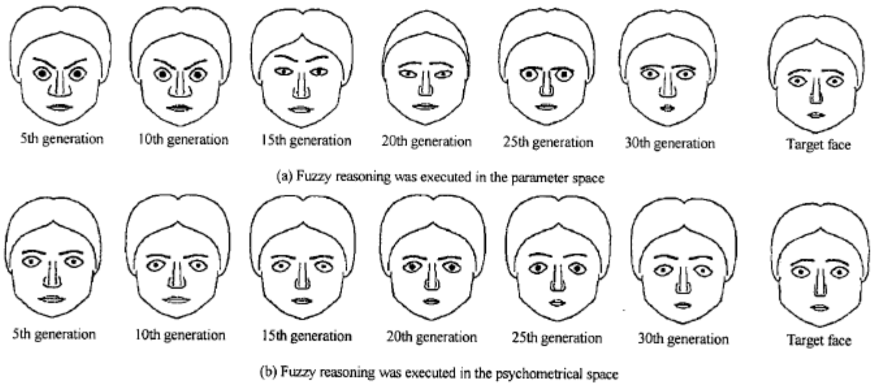

Figure 8: Fuzzy fitness assignment performed in the psychometrical space (lower series) yields really more consistent runs than when run in the parameter space (top series). There are no visual discontinuities between successive generations.

are the coordinates in the parameter space (genotype) and

$$S = (s_1, s_2, ..., s_n)$$

the coordinates in the psychometrical space. This mapping was constructed using subjective experiments and statistical tools (following [14]). First of all, a set of random individuals, enough to represent to whole search space, is generated. Next, human subjects are asked to make similarity groupings among these individuals. This grouping data is turned into a similarity matrix (see [32]) which is fed to a method developed by Kruskal and called "M-D-Scal" (see [19]) that outputs the transformation matrix between F and S.
Experiments are conducted using the traditional face montage system and compare three systems : in the first one, the human operator has to rate every single individual (rating all method), the second one uses fuzzy fitness assignment in the parameter space and the last one uses fuzzy fitness assignment in the psychometrical space obtained by mapping. The first system is immediately taken down as way too heavy to operate : subjects lose their concentrations and results are incoherent. The other two behave properly but the last one, using the psychometrical space shows a really better robustness and coherence along its run. Transitions between generations are way smoother from the perceptive standpoint(see figure 8) and comparisons between runs made by different subjects besides show that the method is more consistent and robust against instability of sensory judgement and variations among subjects.

6.3.2 Factor spaces

Even if application oriented, this second work ([37]) deserves to mentioned here for going a bit further in complexity for the construction of the psychological space. The goal of this research is to design a multimedia database retrieval system based on an interactive GA where the user inputs a couple of keywords in psychological terms (e.g. "gloomy", "passionate", "simple", etc.) and gets

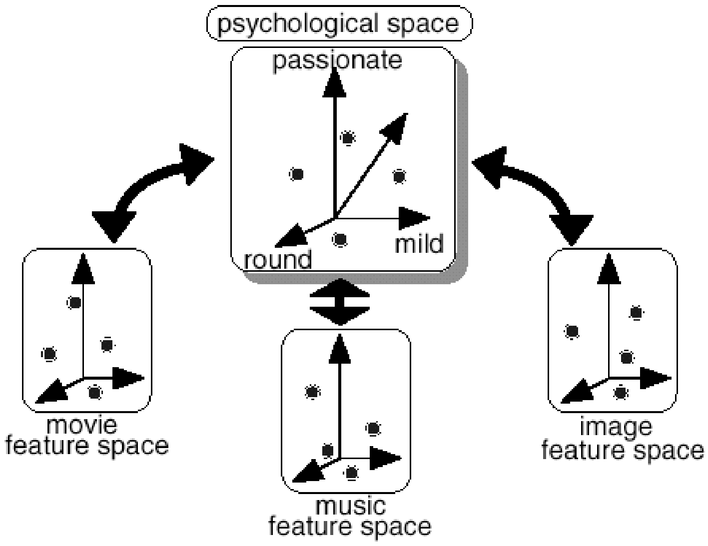

Figure 9: Mappings are constructed between the different feature spaces and the common factor space.

all the corresponding media (images, music pieces, etc.). And to achieve this, a psychometrical space is derived that describes human impressions in a way that is completely independent from the physical characteristics of the object, source of those impressions, may it be a picture, a sound or anything else. This again implies the use of mapping techniques between the feature spaces (one per type of media) and a common psychometrical space (the factor space) as illustrated by figure 9 so that the GA can browse a unique topology that is relevant from the user's point of view and independent of the individuals' material contingencies.

This factor space is constructed using the following procedure :
First of all a set of general adjectives are collected to describe the media (e.g. "bright", "dark", "complex", "serious", etc.) are gathered and grouped in opposite pairs (e.g. "bright" vs "dim, "simple" vs "complex", etc.) these pairs (around 14 in [37]) form the basis of a first description space : the adjective space.
Second of all, all the media present in the database are rated by human subjects in terms of the adjective space.
Next, as this adjective space is way too big to be conveniently manipulated and bears a high degree of redundancy, Principal Component Analysis is performed on the rating data to obtain a compressed space of predefined dimension (usually 3 to 6) : the factor space.
Mappings can now be derived between the feature spaces and this factor space using, for example, an Artificial Neural Network. This mapping can then be used to browse the database with the interactive GA which follows the user's psychological topology and is able to retrieve the corresponding individuals,

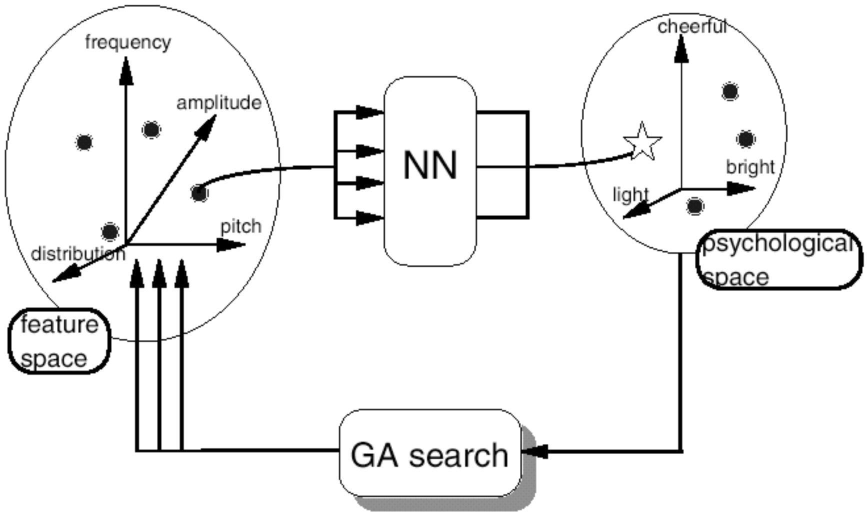

Figure 10: Using the mapping, represented by a Neural Network, the interactive GA is able to browse the database (parameter space) using subjective descriptions (psychometrical space.)

spread all over the database. See figure 10.

Overall, this research suggests that IEC works way better when the user's subjective way to see the world is taken into account.

7 Summary and Conclusions

There are two ways in which an Evolutionary Algorithm can be interactive : a human operator can either provide fitness evaluations or help the genetic search itself. In both cases something can be done to improve the interaction between the GA and the user.

It was first outlined how mathematical description and understanding of IEC is possible in the framework of stochastic automata.

Suggestions came then on how the Graphical Interface can be ameliorated so that fitness evaluations can be quickly and easily provided. Two examples were given, the first one setting an appropriate granularity for fitness evaluation, the second one performing fitness prediction. Two methods of active user intervention were also given. These methods are two examples of how a human being can help improving the quality of the genetic search : by embedding her/his knowledge of the problem and by visualizing the fitness landscape and accordingly guiding the search.

Three general methods of GA enhancement were finally shown to be well adapted to Interactive GAs : hybridization with a local solver, clustered oriented search and the derivation of psychometrical spaces.

It was overall hopefully shown that a competent practice of Interactive Evolutionary Computation is possible and could yield appreciable results in terms of speed, functionality and quality of the search.

References

[1] P.J. Bentley and U.-M. O'Reilly. Ten steps to make a perfect creative evolutionary design system. *Proceedings of GECCO*, 2001.

[2] F. Boschetti and H. Takagi. Visualization of ec landscape to accelerate ec conversion and evaluation of its effect. *Congress on Evolutionary Computation (CEC2001), Seoul, Korea.*, 2001.

[3] C. L. Bridges and D. E. Goldberg. An analysis of reproduction and crossover in a binary-coded genetic algorithm. *Genetic algorithms and their application: Proceedings of the Second International Conference on Genetic Algorithms*, 1987.

[4] C. Caldwell and V.S. Johnston. Tracking a criminal suspect through face-space with a genetic algorithm. *4th International Conference on Genetic Algorithms (ICGA'91), San Diego, CA.*, 1991.

[5] P. D. Cuesta and al. Practical modeling of simple genetic algorithms, via deterministic paths, by absorbing markov chains. *Proceedings of GECCO 2000*, 2000.

[6] R. Dawkins. *The Blind Watchmaker*. Essex, U.K.:Longman, 1986.

[7] K.A. De Jong. *An analysis of the behavior of a calaa of genetic adaptive systems*. PhD thesis, University of Michigan, 1975.

[8] D. E. Goldberg. *Genetic algorithms and simulated annealing*, chapter Simple genetic algorithms and the minimal, deceptive problem. London: Pitman., 1987.

[9] D. E. Goldberg and C. L. Bridges. An analysis of a reordering operator on a ga-hard problem. *Biological Cybernetics*, 62, 1990.

[10] D. E. Goldberg and P. Segrest. Finite markov chain analysis with genetic algorithms. *Proceedings of the 1987 International Conference on Genetic Algorithms and Their Application*, 1987.

[11] N. Hayashida and H. Takagi. Visualized iec : Interactive evolutionary computation with multidimensional data visualization. *Industrial Electronics, Control and Instrumentation (IECON2000), Nagoya Japan.*, 2000.

[12] M. Herdy. Evolution strategies with subjective selection. *International Conference on Evolutionary Computation (ICEC/PPSN IV), Berlin, Germany.*, 1996.

[13] M. Herdy. Evolutionary optimization based on subjective selection - evolving blends of coffee. *5th European Congress on Intelligent Techniques and Soft Computing EUFIT'97, Aachen, Germany.*, pages 640–644, 1997.

[14] N. Honda and F. Sugimoto. Multivariate data classification by face pattern considering psychometrical distance of facial expression/. *Behaviormetrika*, (21):29–43, 1987.

[15] T. Ingu and H. Takagi. Accelerating ga convergence by fitting a single-peak function. *IEEE International Conference on Fuzzy Systems, Seoul, Korea.*, 1999.

[16] Kemeny and Snell. *Finite Markov chains*. Princeton: Van Nostrand Rheinhold.

[17] H.S. Kim and S.B. Cho. An efficient genetic algorithm with less fitness evaluation by clustering. pages 887–894. Proceedings of the 2001 IEEE Congress on Evolutionary Computation, Seoul, Korea, 2001.

[18] T. Kohonen. *Self Organizing Maps*. Springer Verlag, Heidelberg., 1995.

[19] J.B. Kruskal. Nonmetric multidimensional scaling; a numerical method. *Psychometrika*, (29):115–129, 1072.

[20] J.Y. Lee and S.B. Cho. Sparse fitness evaluation for reducing user burden in interactive genetic algorithm. pages 998–1003. Proceedings of the 1999 IEEE Congress on Fuzzy Systems, Seoul, Korea, 1999.

[21] A.E. Nix and M.D. Vose. Modelling genetic algorithms with markov chains. *Annals of Mathematics and Artificial Intelligence*, 1992.

[22] M. Ohsaki and H. Takagi. Improvement of presenting interface by predicting the evaluation order to reduce the burden of human interactive ec operators. pages 1284–1289. Proceedings of the 1998 IEEE International Conference in System, Man and Cybernetics (SMC'98), San Diego, CA, 1998.

[23] U.-M. O'Reilly and Ramachandran G. *A preliminary Investigation of Evolution as a Form Design Strategy - Artificial Life VI*, pages 443–447. MIT PRESS, 1998.

[24] I.C. Parmee. *Evolutionary and Adaptive Computing in Engineering Design*. Springer Verlag, 2001.

[25] A. Paz. *Introduction to probabilistic automata*. Academic Press, New York and London, 1971.

[26] M. Rattray. *Modelling the Dynamics of Genetic Algorithms using Statisitical Mechanics (PhD Dissertation)*. PhD thesis, Manchester University, UK., 1996.

[27] F. Raynal, E. Lutton, P. Collet, and Schoenauer M. Manipulation of non-linear ifs attractors using genetic programming. *Congress on Evolutionary Computation (CEC'99), Washington, D.C.*, 1999.

[28] G. Rudolph. On interactive evolutionary algorithms and stochastic mealy automata. pages 218–226, 1196.

[29] J.D. Schaffer, L.J. Caruana, R.A.and Eshelman, and R. Das. A study of control parameters affecting online performance of gas for function optimization. In *3rd Inernational Conference on Genetic Algorithms (ICGA'89)*, San Mateo, CA, 1989. Morgan Kaufmann Publishers.

[30] Y. Semet, E. Lutton, and P. Collet. Artificial ant colonies and e-learning : optimization of pedagogical path. Submitted to HCI'03, 2002.

[31] Y. Semet, U.-M. O'Reilly, and F. Durand. Interactive artificial ant colonies and evolutionary computation for non photorealistic rendering. Work in Progress, 2002.

[32] R.N. et al.(Eds.) Shepard. *Multidimensional Scaling; Theory and Application in the Behavioural Sciences*, volume 2. Seminar Press, 1972.

[33] K. Sims. Artificial evolution for computer graphics. *ACM SIGGRAPH Conference Proceedings*, 1991.

[34] F. Sugimoto and M. Yoneyama. Robustness against instability of sensory judgment in a human interface to draw a facial image using a psychometrical space model. *IEEE International Conference on Multimedia and Expo. (ICME'2000), New York, New York.*, 2000.

[35] H. Takagi. Active user intervention in an ec search. *5th joint Conference on Information Sciences (JCIS2000), Atlantic City, NJ.*, pages 995–998, 2000.

[36] H. Takagi. Interactive evolutionary computation: fusion of the capabilities of ec optimization and human evaluation. *Proceedings of the IEEE*, 89:1275 – 1296, Sept. 2001.

[37] H. Takagi and Kishi K. On-line knowledge embedding for an interactive ec-based montage system. *3rd International Conference on Knowledge Based Intelligent Information Engineering Systems (KES'99). Adelaide, Australia.*, pages 280–283, 1999.

[38] H. Takagi and K. Ohya. Discrete fitness values for improving the human interface in an interactive GA. pages 109–112, Piscataway, NJ, 1996. IEEE Service Center.

[39] T. Unemi. Sbart 2.4 : Breeding 2d cg images and movies and creating a type of collage. *3rd International Conference on Knowledge-Based Intelligent Information Engineering Systems (KES799), Adelaide, Australia.*, 1999.