R INRIA

Visual analytics and experimental analysis of evolutionary algorithms

Evelyne Lutton — Jean-Daniel Fekete

N° 7605

April 2011

_____ Interaction and Visualization _____

R apport de recherche

Visual analytics and experimental analysis of evolutionary algorithms

Evelyne Lutton , Jean-Daniel Fekete

Theme : Interaction and Visualization
Équipe-Projet aviz

Abstract: Experimental analysis of evolutionary algorithms usually aims at tuning the parameter setting or at improving knowledge about internal mechanisms (operators efficiency, genetic material distribution, or diversity management for instance). This crucial step relies on the analysis of a huge amount of multidimensional data, including numeric and symbolic data. Usual features of existing EA visualisation systems consist in visualising time- or generation-dependent curves (fitness, diversity, or other statistics). But when dealing with detailed genomic information, the task becomes more difficult, as a convenient visualisation strongly depends on the considered fitness landscape. In this latter case the raw data are usually sets of successive populations of points of a complex multidimensional space. The purpose of this paper is to evaluate the potential interest of some recent visual analytics tools for navigating in complex sets of EA data, and to sketch future developements of visual analytics tools adapted to the needs of EA experimental analysis.

Key-words: Optimisation, Artificial Evolution, Genetic algorithms, Visual Analytics, Experimental analysis of algorithms, parameter tuning, fitness landscape visualisation

Visualisation interactive et analyse expérimentale d'algorithmes évolutionnaires

Résumé : L'analyse expérimentale des algorithmes évolutionnaires nécessite d'appréhender une grande quantité de données multidimensionelles, incluant le plus souvent des données numériques et symboliques. Il est extrêmement difficile de naviguer dans un tel ensemble de données, et plus encore d'en tirer des conclusions permettant par exemple de mesurer l'efficacité des différents opérateurs ou de régler les paramètres de l'algorithme. Les outils de visualisation qui accompagnent les différentes boîtes à outils d'algorithmes évolutionnaires donnent accès la plupart du temps à des courbes d'évolution en fonction du temps ou de la génération (fonction de fitness, indicateurs de diversité, ou autres statistiques). En ce qui concerne les informations génomiques, la tâche devient beaucoup plus ardue, car la visualisation dépend fortement du paysage de recherche considéré. Dans ce dernier cas les informations brutes se présentent sous forme d'une succession de populations de points définis dans un espace complexe et multidimensionel. Le but du travail présenté ici est d'évaluer le potentiel des outils génériques de visualisation dans le cadre de l'analyse d'algorithmes évolutionnaires, et de proposer différentes pistes de développement pour mieux adapter ces outils aux besoins de la communauté évolutionnaire.

Mots-clés : Optimisation, Evolution artificielle, Algorithmes génétiques, Analyse expérimentale d'algorithmes, Visualisation, Réglage de paramètres, Visualisation de paysages d'optimisation

1 Introduction

The crucial problem of parameter tuning and the huge amount of information produced by Evolution Algorithms (EAs) has led authors to propose various visualisation software adapted to on-line visualisation, that displays fitness curves, convergence diagrams, and some statistics. The important question is to be able to delve into what happens internally during or after the run of an EA. According to Shine and Eick [1] visualization of GAs can be helpful in analyzing the extent to which a GA explores the search space, analyzing its convergence behavior, enabling the user to get some feeling for the fitness landscape and the dynamics of the evolutionary process, and fine-tuning the GA.

The methods proposed are commonly split into two types of visualisation tools: on-line tools, that display a set of monitoring curves during an EA run (fitness of the best individual, statistics, diversity, for instance), and off-line tools, that do a "post-mortem" analysis and provide more sophisticated analyses, including important utilities like visualisation of genomes, and fitness landscapes analysis tools using 2D, 3D plots, density images, graphs, and providing various other representations.

When reviewing the literature on this subject or experimenting existing software and toolboxes, a series of important issues seems to remain difficult:

- How to visualise an individual. This task is particularly complex when the search space is large and multidimensional, when the genome combines symbolic, and numeric (discrete and/or continuous) values. The particular case of genetic programming has also been considered[2]. Proposed solution are usually problem dependent, and may sometimes necessitate the display of phenotypes (signals, images, sounds, graphs, networks, etc ...).

- How to visualise a population, i.e. a set of multidimensional points that may be large. Being able to efficiently visualise fitness landscapes is still a challenge, for which approaches like fitness-distance correlations are only a partial answer.

- How to visualise the history and evolutionary mechanisms. This issue is important, as visualising various statistics about an evolving population may not be enough to understand some of the complex mechanisms of EAs, like the action of operators. Being able to follow the transmission of genetic material inside a population has been partially addressed by schemata analysis, there is a strong need however when dealing with continuous landscapes.

- How to visualise the result in non-standard EAs. A good example of this issue is the case of multi-objective EAs, as the growing size of the problem EA are able to solve lead to outputs made of large Pareto datasets with many objectives. There is a strong need of efficient visualisation tools, that may help to monitor (on line or even interactively) multidimensional Pareto fronts [3].

In this paper, we investigate some recent tools developed by the visual analytics community, that may provide efficient and generic answers to the previous challenges. The paper is organised as follows. A short review of existing visualisation systems for EAs is given in section 2. An analysis of population clouds using Scatter-Dice/GraphDice is developed in section 3 for some classical test-functions. Tests are based on the EASEA language[4][1]. Then a genealogy analysis using GeneaQuilts is

---

[1]The software is available at `http://sourceforge.net/projects/easea/`

presented in section 4. Section 5 will draw a conclusion and sketch future work we intend to develop on this topic.

2 Related work

2.1 On-line visualisation

Almost every evolutionary software proposes nowaday its own on-line visualisation facilities. It is often reduced to the visualisation of best fitness evolution along generations. We give below some examples that provide additional features.

Bedau and Bullock[5, 6] show the importance of tracking evolutionary activity via plots that visualise the history at different levels. For instance, genotype's activity corresponds to the frequence of a given genotype in a population, which appears, increases or decreases along generations, forming what they call "waves".

Pohlheim[7, 8] proposed in 1999 a visualisation adapted to the Genetic and Evolutionary Algorithm Toolbox for Matlab - GEATbx[9]. His system allows a visualisation of the data in various ways, and gives for instance the current state of a population (one generation), visualises a run (all the generations), or different runs for comparisons. Additionally he copes with the problem of visualising high-dimensional data using multidimensional scaling (reduction to a 2D visualisation that preserves distances relationships), and uses a 2D representation to show paths followed by the best individual of every generation.

Kerren and Egger in 2005 [10] developed a java-based on-line visualization tool, EAVis, in which it is possible to embed an application-dependent Phenotype View.

Collins in 2003 [11] provided a survey chapter on this topic and identified some directions for future research. His main conclusion concerns the strong need for flexible visualisation environments, as he considered that current solutions were still too problem dependent.

2.2 Off-line visualisation

Off-line visualization systems allow displaying more data, including multidimensional data, which is one of the important issues in current visualisation systems. Spears [12] provided in 1999 a brief overview of multidimensional visualization techniques for visualizing EAs, such as the use of color, glyphs, parallel coordinates or coordinates projections. For discrete data, Routen in 1994 [13] suggested adopting a modified Hinton diagram[2] in which a rectangle represents a chromosome and its size is determined by the fitness of the chromosome. Let us give below a list of some off-line systems:

- William Shine and Christoph Eick[1] describe the features of a GA-visualization environment that uses quadcodes to generate search space coverage maps, that employs 2D-distance maps to visualize convergence, and uses contour maps to visualize fitness.

- The VIS system [14] proposed in 1999 allows a navigation at various levels of detail, and manages transitions between related data levels. The visualisation of the most detailed level is based on ad-hoc representations (bar codes, colors,

---

[2]A Hinton diagram provides a qualitative display of the values in a matrix. Each value is represented by a square whose size is related to the magnitude, and whose color indicates the sign.

alleles frequencies) but does not allow visualising multidimensional continuous genomes.

- Emma Hart [15] proposed GAVEL in 2001, an off-line visualisation system adapted to generational GAs, that provides a means to understand how crossover and mutation operations assemble the optimal solution, and a way to trace the history of user-selected sets of allele. It allows a display of the complete history across all generations of chromosomes, individual genes, and schemas.

- Marian Mach [16] presented in 2002 a simple and interactive "post-mortem" GA visualising tool focussed on the visualisation of multidimensional data via 1D projections.

Annie Wu [17], who developed the VIS system, gave the following list of desirable tasks for visualisation systems: *(a)* to examine individuals and their encodings in detail, *(b)* to trace the source and survival of building blocks or partial solutions, *(c)* to trace family trees to examine the effects of genetic operators, *(d)* to examine populations for convergence, speciation, etc, *(e)* to trace gross population statistics and trends to move freely in time and through populations.

The genome representation issue, item *(a)*, is perhaps the most complex one, and as we have seen above various solutions have been proposed, depending if we are dealing with continuous or discrete genomes. For Genetic Programming, the question is even more complex: a solution proposed by Jason Daida [2] in 2005 consists in visualizing big tree structures as large sized graphs.

The issue addressed by the GAVEL system, that appears as very challenging in off-line systems and that is mentioned by Annie Wu as items *(c)* and *(d)*, is to be able to trace history of individuals. Spears [12] mentioned also that being able to track the origin of the fittest individual per generation is a very important issue for parameters tuning.

The question of family trees visualisation has more recently been considered by Zbigniew Walczak in 2005 in a short chapter [18] where he proposed to visualise evolutionary processes using graph drawing software.

2.3   Visualisation and interaction

There also exist attemps to use visual analytics in interactive evolution [19, 20] that has been considered as a way to reduce user fatigue. In [19], the system uses visual analytics to display past searched individuals to the user, allowing him to decide where to search. Hayashida and Takagi have shown that such an augmented interaction allow a faster convergence, and expect this method to be powerful in difficult IEC tasks. Llora & al, [20] propose to use a simple visual analytics tool to mine various models of user preferences, allowing them to generate what they call educated guesses or promising solutions.

3   Analysing populations clouds using ScatterDice/GraphDice

Visual analytics, says Wikipedia, is "the science of analytical reasoning facilitated by visual interactive interfaces." This multidisciplinary field integrates various sophisticated computational tools with innovative interactive techniques and visual representations to facilitate human interpretation, or in other terms, the sense-making process

from raw data. Some tools developed in this field could be very useful for the EA community. We present in this section and the next one, two visualisation tools that seem to answer in a generic way to some needs identified in the previous section.

ScatterDice[21] is a multidimensional visual exploration tool, that allows the user to navigate in a multidimensional set via simple 2D projections, organised as scatterplot matrices, visual coherence between various projections is based on animated 3D transitions. A scatterplot matrix presents an overview of the possible configurations, thumbnails of the scatterplots, and support for interactive navigation in the multidimensional space. Various queries can be built using bounding volumes in the dataset, sculpting the query from different viewpoints to become more and more refined. Furthermore, the dimensions in the navigation space can be reordered, manually or automatically, to highlight salient correlations and differences among them[3].

A recent evolution of ScatterDice using the same principles but with many additional features is GraphDice[22]. It allows reading the same type of data (.csv files), and other more sophisticated formats, as it also embeds graph visualisation utilities[4].

This tool can be used to visualise data collected during the run of an EA. At each generation, the composition of the current population can be written in a ".csv" file as on figure 1, forming what can be called a "cloud" of successive populations. This cloud of multidimensional points is visualised using ScatterDice or GraphDice, to produce various, sometimes unusual, viewpoints.

```
Generation;Fitness;x[0];sigma[0];x[1];sigma[1]
INT;DOUBLE;DOUBLE;DOUBLE;DOUBLE;DOUBLE
0;3.75447;-0.12508;0.195626;0.524069;0.255402
0;1.17484;-0.573358;0.142053;0.924887;0.392851
0;2.28066;-0.533583;0.183641;0.546523;0.461643
0;1.92319;-0.70052;0.338246;0.582533;0.406443
0;2.75539;0.784538;0.182157;-0.940648;0.383136
0;3.08894;-0.770051;0.190012;-0.840338;0.359992
0;2.30766;0.380979;0.124581;0.0379446;0.469388
0;3.30957;-0.704403;0.453222;0.208484;0.182612
...
```

Figure 1: A simple .csv file collected during a run (minimisation of the 2D Weierstrass function $H = 0.2$)

The test functions used in the experiments below are the following:

- Weierstrass functions (see figure 2) defined in a space of dimension 2, of Hölder exponents $H = 0.2$ (very irregular) and $H = 0.9$ (more regular).

$$f(x, y) = \sum_{n=-\infty}^{+\infty} 2^{-nH}(1 - \cos 2^n x) + \sum_{n=-\infty}^{+\infty} 2^{-nH}(1 - \cos 2^n y)$$

- Rosenbrock function[5] in a space of dimension 10.

$$f(x_1, ..., x_{10}) = \sum_{n=1}^{9} 100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2$$

---

[3] A demo of ScatterDice can be launched from `http://www.aviz.fr/~fekete/scatterdice/`, it accepts standard .csv files (although it may be necessary to add a second line after the header giving the data type for each column - INT, STR, REAL, etc).

[4] A demo of ScatterDice is also accessible at `http://www.aviz.fr/graphdice/`

[5] See for instance http://en.wikipedia.org/wiki/Rosenbrock_function

$H = 0.2$ (very irregular) $H = 0.9$ (more regular).
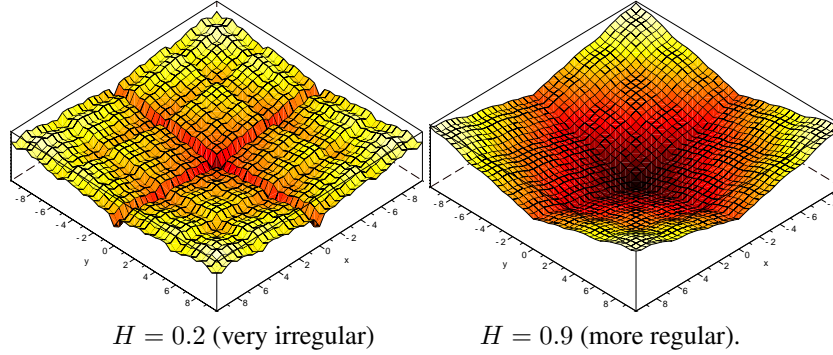
Figure 2: 2D Weierstrass functions with Hölder exponent H.

The genetic engine is a simple generational algorithm on R^n using tournament selection, geometric (barycentric) crossover and log-normal auto-adaptative mutation. Additional dimensions are thus considered in the search space, the σ_i values, that represent the mutation radius for each coordinate x_i. The population size is 100 and the algorithm runs for 100 generations. This genetic engine is available in the sample programs (weierstrass.ez) distributed with the EASEA software.

A visualisation of the population cloud for the 2D Weierstrass function of dimension 0.2 is given on figure 3. The scatterplot matrix on the left of the figure, gives an overview of the possible visualisations. The columns and lines of this matrix can be dragged and dropped as wished by the user, a default order is proposed, based on an algorithm that reduces "clutter"[22]. On the right, a detailed view is given, on which some queries have been visualised: in red, the points corresponding to the 10 first generations of the run, in yellow, the 10 last generations, and in green, the best fitness points. The queries are organised as layers, the "Query layers" window gives the details of the three queries, with some additional measurements (percentage of selected points, and percentage of selected edges if a graph is visualised : GraphDice actually considers a set of points as a degenerate graph, made of a collection of nodes with no edges). Bottom left, a "Selection History" window show how the queries have been sculpted: queries 1 and 2 have been activated on the Generation versus Generation view, i.e. the top left plot of the scatterplot matrix, while query 3 has been made on the fitness versus generation plot. On the extreme rigth of the window, a toolbar proposes various visualisation options (the "Show" window), for instance "Grid" activates a grid on the dataset, "Labels" allow to display attributes attached to a point. It is thus for instance possible to identify different runs of an EA using a label, and use this option to separate the data when needed. "Hull" displays a convex hull for each query, and "Zoom" activates an automatic zoom focussed on the selected data.

The Fitness versus Generation plot, that displays the whole set of individuals generated along the evolution, provides additional information about the distribution of successive populations, for comparison figure 4 gives the classical fitness evolution curves. When observed from a different viewpoint, like in figure 5, it can be noticed that the population diversity decreases slowly, and converge toward a point of the 2D plane, while stabilising on some mutation parameters. The first generations (red points of the query 1) are spread in a rather uniform way on the whole search space, while the last ones (yellow points) are concentrated in the areas of best fitness (green points).
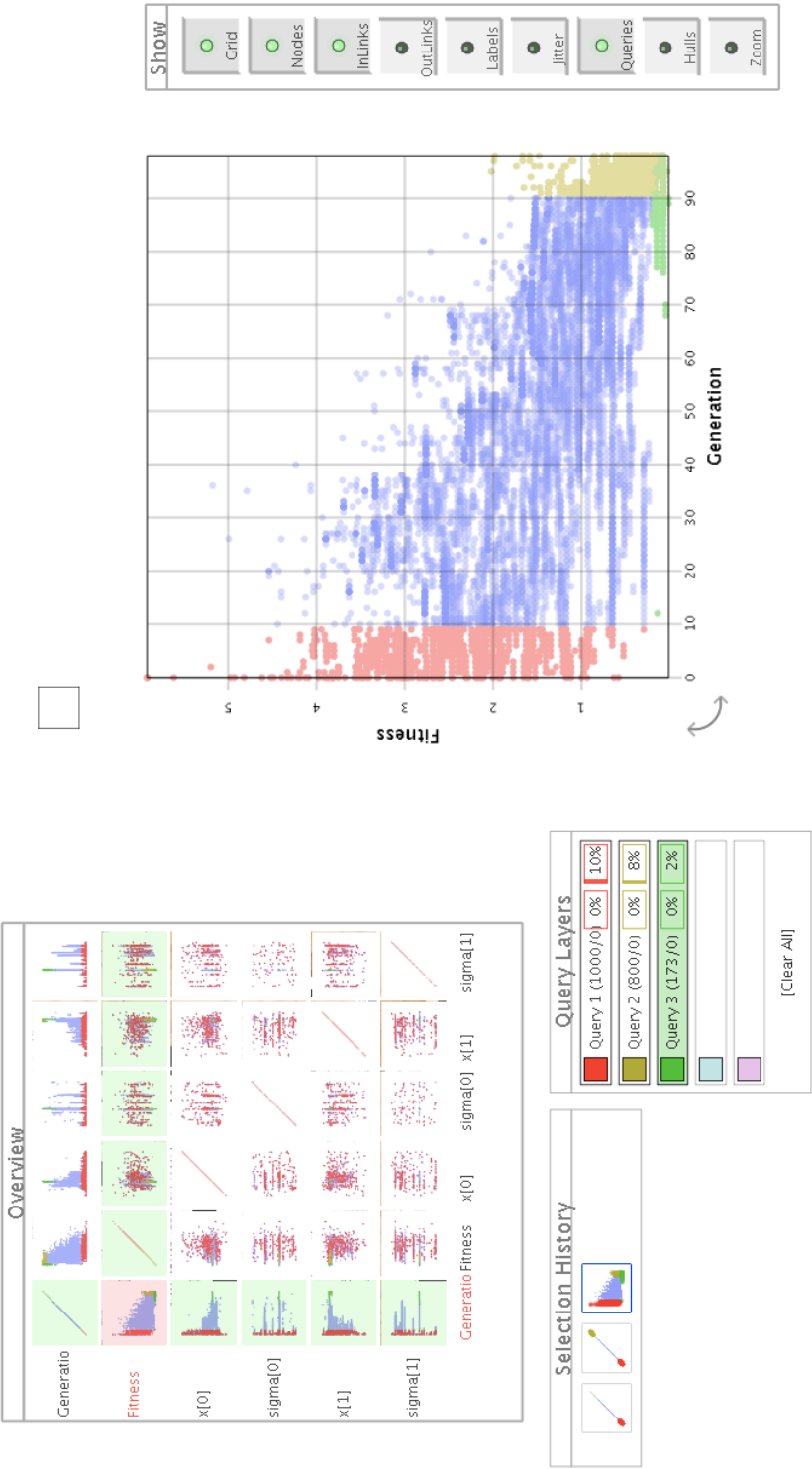
Figure 3: 2D Weierstrass function of Hölder exponent 0.2. Scatterplot and Fitness versus generation view. Red points correspond to the first 10 generations, yellow points, to the 10 last ones, and green points, to the best fitness areas.
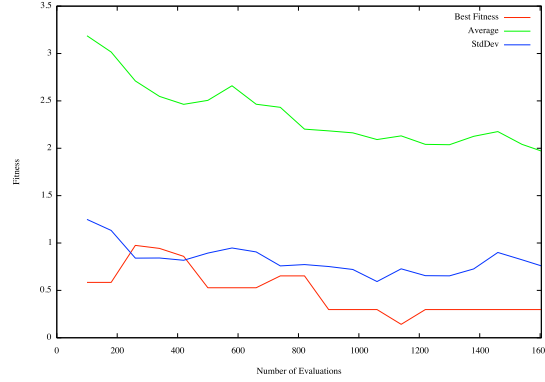
Figure 4: 2D Weierstrass function $H = 0.2$, classical best fitness, average fitness curves

A green point appears rather early (see generation 12 on the Fitness versus Generation plot), but green points start to multiplicate rather late (from generation 68) in the evolution.
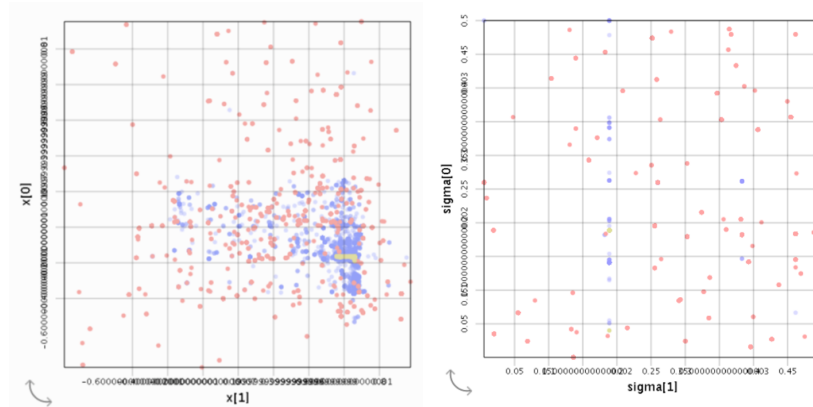
Figure 5: 2D Weierstrass function $H = 0.2$. Left, projection on the 2D plane (x_0, x_1). Right, parameters σ_0 and σ_1. Points in red belong to generations 0 to 10, points in yellow to 90 to 100.

Figure 6 gives the same view as figure 3 but for a 2D Weierstrass function of dimension 0.9. This function is much more regular than the previous one. Visually, it seems obvious that the population is able to converge more rapidly.

Figure 7 gives an overview of the visualisation window for a dimension 10 space. Once again, as the function is more regular, the population seems to converge rapidly, even if it uses the same parameters as for the Weierstrass functions, in a search space of larger dimension.
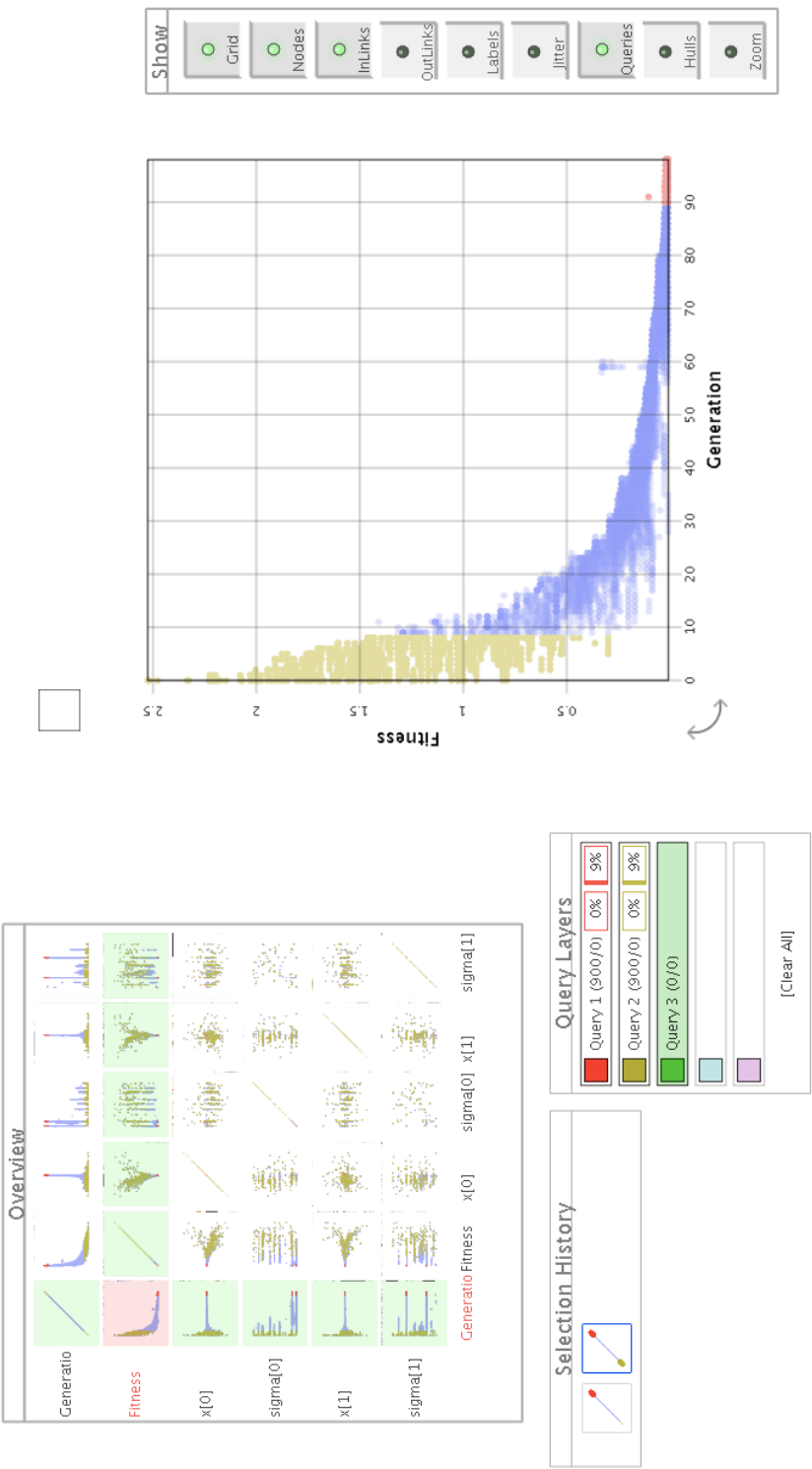
Figure 6: 2D Weierstrass function of Hölder exponent 0.9. Yellow points correspond to the first generations, red points to the last generations.
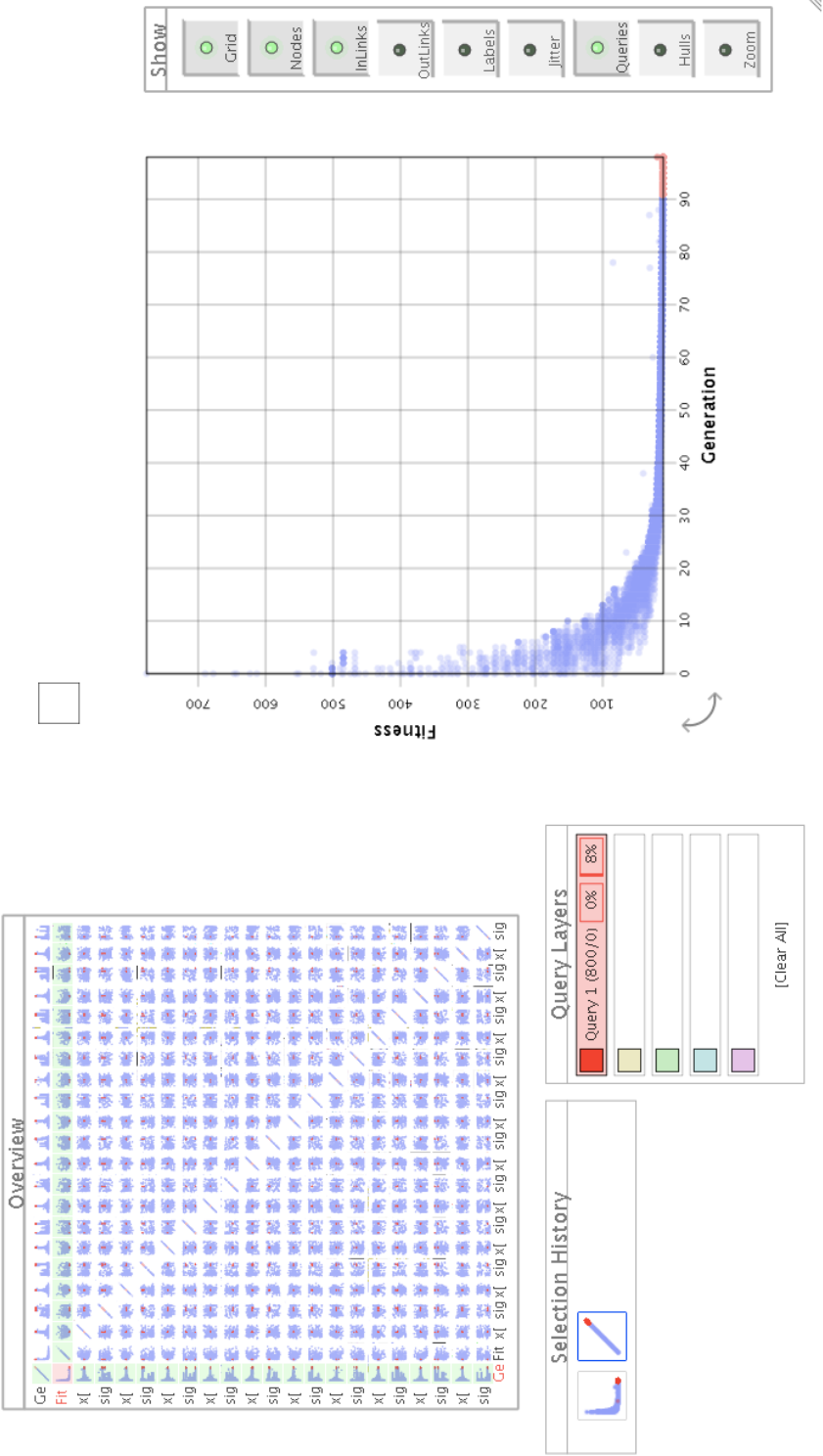
Figure 7: Rosenbrock function in dimension 10. Red points correspond to the 10 last generations.

4 Analysing EA genealogies using GeneaQuilts

Another challenge for EA visualisation systems is to provide a versatile visualisation of EAs history, and of the behaviour of genetic operators. We show below that visualisation systems developed for genealogist have interesting features that can be adapted to the needs of EAs visualisation.

Genealogy research actually relies more and more on sophisticated computer systems, as current genealogies databases can easily reach tens of thousands of nodes. For visualisation, node-link diagrams are the most widespread representations, but it has been shown such diagrams quickly become unreadable as graph size grows. More scalable visualisation solutions, based on Quilts representation, have been recently proposed[23]. Quilts visualisation displays layered graphs in a more compact way, as it eliminate the confusing link crossings of node-links diagrams.

This visualisation takes the form of a diagonally-filled matrix, where rows are individuals and columns are nuclear families. Figure 8 gives an example of a small family quilt display. The Simpson family is organised in 3 generations, from left (the oldest) to right (the youngest). Columns correspond to families and are identified by an "F" icon, in which round and square dots respectively correspond to male and female. For instance, Homer and Marge are the parents of Bart, Maggie and Lisa (rightmost "F" column). Then, left to Homer and Marge, the two columns allow knowing who are their parents and siblings.

The GeneaQuilts system includes an overview, a timeline, search and filtering components, and a new interaction technique called Bring & Slide that allows fluid navigation in very large genealogies.[6]
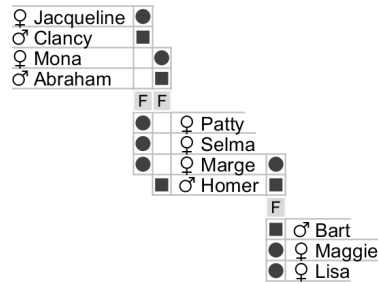

Figure 8: GeneaQuilts Visualization of the Simpson Family (from [23])

Experiments below have been performed on the same set of test functions as in section 3. Data are collected as text and transformed into ged format[7] via a simple parse routine, see figure 9.

Figure 10 shows the GeneaQuilts window: on the top is represented a time-line view, on the top right an overview of the whole genealogy, and on the bottom right details about a selected individual. The main window gives a zoom, corresponding to the small blue box of the overview (up right), and the ancestors of a selected individual are tracked in red on the main window and on the overview. It is also possible to

---

[6]A demo is available on http://www.aviz.fr/geneaquilts/
[7]See http://en.wikipedia.org/wiki/GEDCOM for details about this format.

```
Individu : NoGen 1 -- id 21 -- fitness 95
Individu : NoGen 1 -- id 22 -- fitness 310
Individu : NoGen 1 -- id 23 -- fitness 270
Individu : NoGen 1 -- id 24 -- fitness 103
Individu : NoGen 1 -- id 25 -- fitness 305
Individu : NoGen 1 -- id 26 -- fitness 290
Individu : NoGen 1 -- id 27 -- fitness 95
Individu : NoGen 1 -- id 28 -- fitness 282
Individu : NoGen 1 -- id 29 -- fitness 275
Individu : NoGen 1 -- id 30 -- fitness 95
Individu : NoGen 1 -- id 31 -- fitness 323
Individu : NoGen 1 -- id 32 -- fitness 326
Individu : NoGen 1 -- id 33 -- fitness 95
Individu : NoGen 1 -- id 34 -- fitness 196
Individu : NoGen 1 -- id 35 -- fitness 65
Cross : NoGen 2 -- parent1 32 -- parent2 32 -- son 36
Cross : NoGen 2 -- parent1 23 -- parent2 32 -- son 37
Cross : NoGen 2 -- parent1 32 -- parent2 32 -- son 38
Cross : NoGen 2 -- parent1 32 -- parent2 13 -- son 39
Cross : NoGen 2 -- parent1 32 -- parent2 32 -- son 40
Cross : NoGen 2 -- parent1 32 -- parent2 32 -- son 41
Mutation : NoGen 2 -- id_parent 32 -- id_mute 42
Cross : NoGen 2 -- parent1 32 -- parent2 35 -- son 43
Cross : NoGen 2 -- parent1 33 -- parent2 7 -- son 44
Cross : NoGen 2 -- parent1 32 -- parent2 17 -- son 45
Cross : NoGen 2 -- parent1 2 -- parent2 30 -- son 46
Cross : NoGen 2 -- parent1 32 -- parent2 32 -- son 47
Cross : NoGen 2 -- parent1 35 -- parent2 17 -- son 48
Cross : NoGen 2 -- parent1 32 -- parent2 32 -- son 49
Individu : NoGen 2 -- id 36 -- fitness 326
Individu : NoGen 2 -- id 37 -- fitness 256
Individu : NoGen 2 -- id 38 -- fitness 326
...
```

```
0 HEAD
1 SOUR ArtiFract
1 FILE genealogie.ged
1 CHAR ANSI
0 @I1@ INDI
1 NAME FIRST_GEN_id1_gen0
1 FITNESS 95
1 GENERATION 0
1 BIRT
2 DATE 1 JAN 0
1 DEAT
2 DATE 1 JUN 1
1 FAMS @F1@
1 FAMS @F1@
1 FAMS @F3@
1 FAMS @F4@
1 FAMS @F5@
1 FAMS @F1@
1 FAMS @F1@
1 FAMS @F7@
1 FAMS @F8@
1 FAMS @F1@
1 FAMS @F1@
1 FAMS @F10@
1 FAMS @F1@
1 FAMS @F1@
0 @I2@ INDI
1 NAME FIRST_GEN_id2_gen0
1 FITNESS 178
1 GENERATION 0
1 BIRT
...
```

Figure 9: Geneational data: simple text data collected during the run (left) and corresponding ged format (right)

track all the offsprings of an individual (figure 10), or to visualise common branches of two lineages (ancestors or offsprings, see figure 11). In order to identify the action of operators, we took as convention that female individuals are created by mutation while male ones are the result of a crossover. We thus have a mean to track the efficiency of operators all along a lineage. There is no limitation of the number of genders in the system, a visualisation of more than two operators is thus possible.

Experiments also show that the genealogy expands in different ways depending on the irregularity of the fitness function. Figure 12 shows two genealogy profiles for two Hölder exponents of the Weierstrass function. The most regular function ($H = 0.9$) has a genealogy that keeps longer alife some individuals, producing a structure with longer inter-generational mixture than for the more irregular function ($H = 0.2$). We conjecture this is linked to the number and size of basins of attraction of local optima.
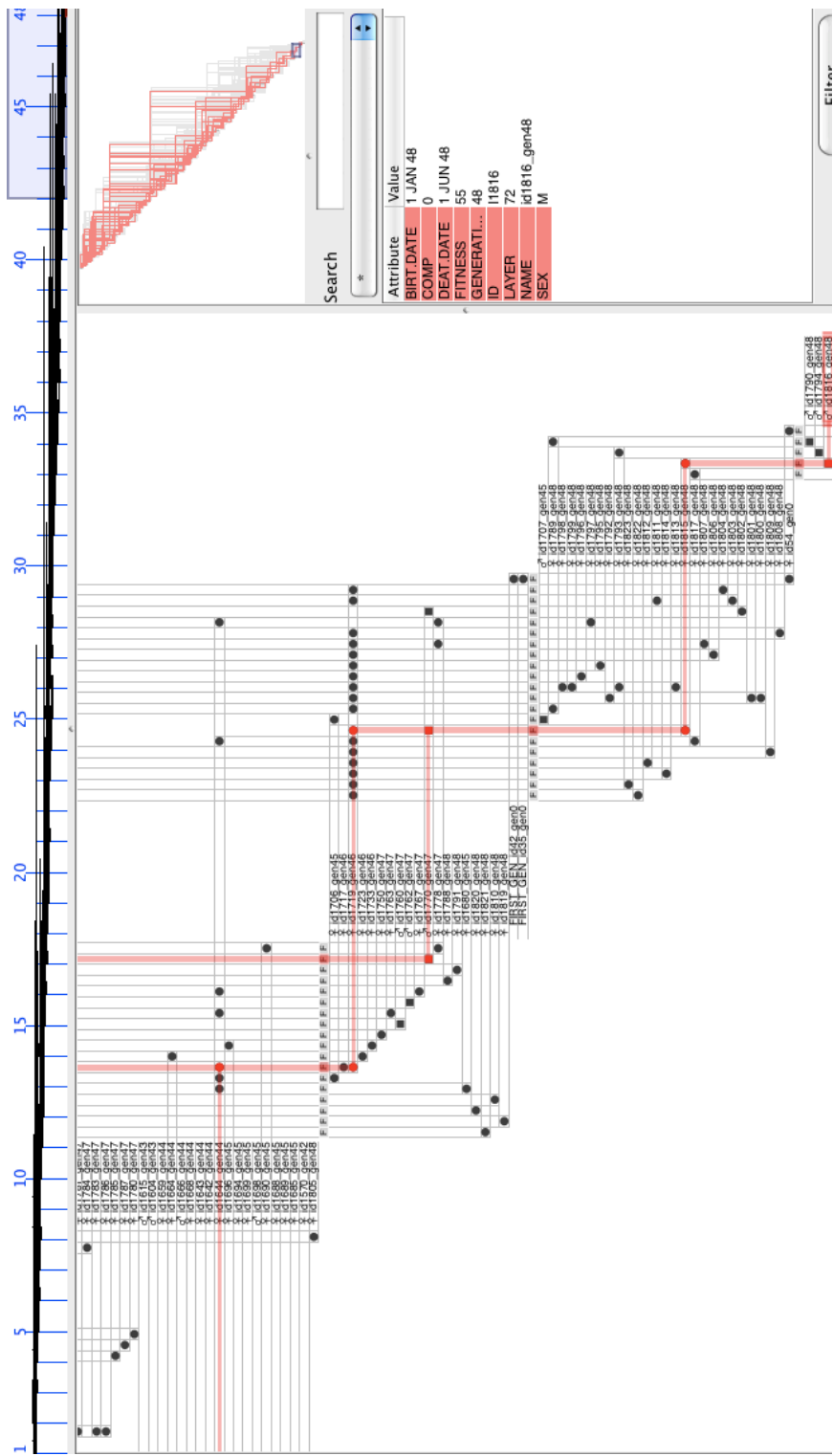
Figure 10: Rosenbrock function in dimension 10 (50 individuals during 50 generations). Zoom on the last generations and track of the ancestor of a good individual (in red)
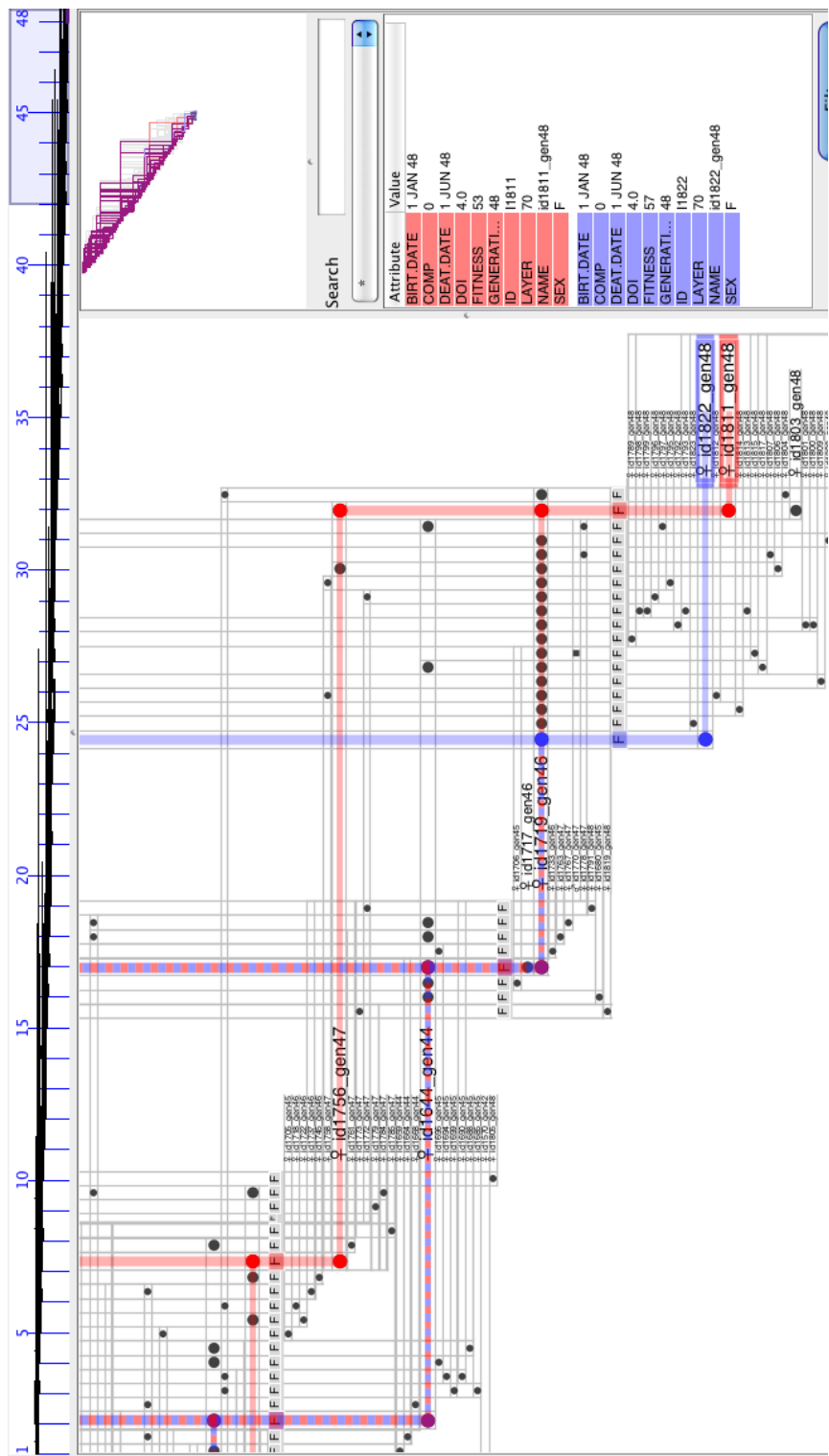
Figure 11: Rosenbrock function in dimension 10 (50 individuals during 50 generations). Track of the common ancestors of two individuals: continuous red and blue lineages until the first common ancestor from which the lineage is identified with a red-blue dotted line). A filtering has been applied, to magnify the selected lineages.

$$H = 0.2$$

$$H = 0.9$$
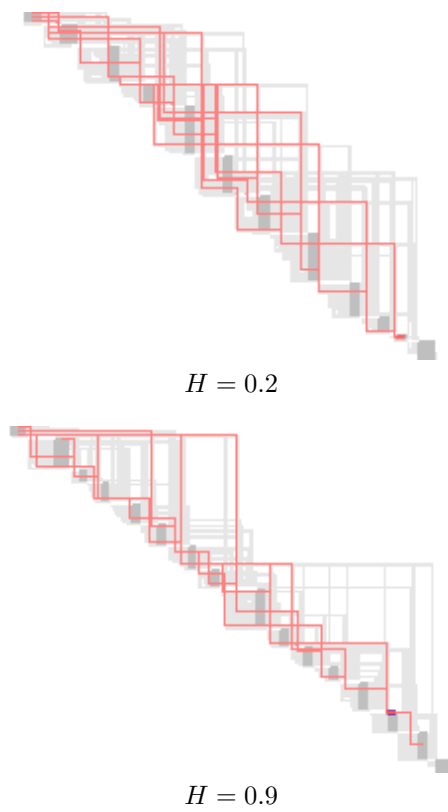
Figure 12: Profile of a genealogy of 20 individuals during 20 generations for Weierstrass functions of exponent 0.2 (top) and 0.9 (bottom). The Lineage of a good individual of the last population is highlighted in red.

5 Conclusion and future work

We have shown above that two visualisation software of the infovis community, GraphDice and GeneaQuilts, can easily be used for visual inspection of EA behaviour. First experiments have been performed on toy problems, but let us identify features that may shed new light on the experimental analysis of EAs. GraphDice for instance allows a rapid visual inspection of exploration capability and population diversity, and GeneaQuilts provides a visualisation of the impact of operators. GeneaQuilts may also allow visualising genetic drift effects, as it is able to track the lineage of two individuals and to identify their common ancestors.

This preliminary study allows identifying a set of desirable features to adapt these general visualisation tools to the specific needs of EA analysis. The following issues will guide future developments of GeneaQuilts and GraphDice versions adapted to EAs:

- Tests have been performed on relatively small data sets (up to 100x100 individuals in 10 dimensional space). The scalability issue will be tested more extensively.

- Various usual statistics (per generation, per fitness level) and query-based statistics will be implemented, including comparison of distributions (p-values).

- For GeneaQuilts, original features like lineage statistics (which may for instance provide various numerical measurements for the efficiency of genetic operators) or similarity measurement by identification of common ancestors will be considered.

- GraphDice, that also allow to visualise graphs, will be evaluated for the genealogy visualisation task.

- On-line visualisation issues will also be considered.

References

[1] Shine, W., Eick, C.: Visualizing the evolution of genetic algorithm search processes. In: Proceedings of 1997 IEEE International Conference on Evolutionary Computation, IEEE Press (1997) 367–372

[2] Daida, J., Hilss, A., Ward, D., Long, S.: Visualizing tree structures in genetic programming. Genetic Programming and Evolvable Machines **6** (2005) 79–110 10.1007/s10710-005-7621-2.

[3] Parmee, I., Abraham, J.: Supporting implicit learning via the visualisation of coga multi-objective data. In: CEC2004, Congress on Evolutionary Computation, 19-23 June. Volume 1. (2004) 395 – 402

[4] Collet, P., Lutton, E., Schoenauer, M., Louchet, J.: Take it EASEA. In Schoenauer, M., Deb, K., Rudolf, G., Yao, X., Lutton, E., J.J., M., Schwefel, H.P., eds.: Parallel Problem Solving from Nature - PPSN VI 6th International Conference, Paris, France, Springer Verlag (September 16-20 2000) LNCS 1917.

[5] Bedau, M.A., Joshi, S., Lillie, B.: Visualizing waves of evolutionary activity of alleles. In: Proceedings of the 1999 GECCO Workshop on Evolutionary Computation Visualization. (1999) 96–98

[6] Bullock, S., Bedau, M.A.: Exploring the dynamics of adaptation with evolutionary activity plots. Artif. Life **12** (March 2006) 193–197

[7] Pohlheim, H.: Visualization of evolutionary algorithms - set of standard techniques and multidimensional visualization. In: Proceedings of the Genetic and Evolutionary Computation Conference, Morgan Kaufmann (1317) 533–540

[8] Pohlheim, H.: Visualization of evolutionary algorithms - set of standard techniques and multidimensional visualization. In: GECCO'99 - Proceedings of the Genetic and Evolutionary Computation Conference, San Francisco, CA: Morgan Kaufmann. (1999) 533–540

[9] Pohlheim, H.: Geatbx - genetic and evolutionary algorithm toolbox for matlab http://www.geatbx.com/.

[10] Computer, A.K., Kerren, A.: Eavis: A visualization tool for evolutionary algorithms. In: Proceedings of the IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 05. (2005) 299–301

[11] Collins, T.D. In: Visualizing evolutionary computation. Springer-Verlag New York, Inc., New York, NY, USA (2003) 95–116

[12] Spears, W.M.: An overview of multidimensional visualization techniques. In: Evolutionary Computation Visualization Workshop. (1999) T. D. Collins, editor, Orlando, Florida, USA.

[13] Routen, T.: Techniques for the visualisation of genetic al- gorithms. In: The First IEEE Conference on Evolutionary Computation. Volume II., IEEE Service Center (1994) 846–851

[14] Wu, A.S., De Jong, K.A., Burke, D.S., Grefenstette, J.J., Ramsey, C.L.: Visual analysis of evolutionary algorithms. In: Congress on Evolutionary Computation (CEC'99), IEEE Press (1999) 1419–1425

[15] Hart, E., Ross, P.: Gavel - a new tool for genetic algorithm visualization. IEEE Trans. Evolutionary Computation **5**(4) (2001) 335–348

[16] Mach, M. Zetakova, Z.: Visualising genetic algorithms: A way through the Labyrinth of search space. In: Intelligent Technologies - Theory and Applications. IOS Press, Amsterdam (2002) 279–285 P. Sincak - J. Vascak - V. Kvasnicka - J. Pospichal (eds.).

[17] Wu, A.S., Jong, K.A.D., Burke, D.S., Grefenstette, J.J., Ramsey, C.L.: Visual analysis of evolutionary algorithms. In: In Proceedings of the 1999 Conference on Evolutionary Computation (CEC'99, IEEE Press (1999) 1419–1425

[18] Walczak, Z.: Graph-based analysis of evolutionary algorithm. In Klopotek, M., Wierzchon, S., Trojanowski, K., eds.: Intelligent Information Processing and Web Mining. Volume 31 of Advances in Soft Computing. Springer Berlin / Heidelberg (2005) 329–338

[19] Hayashida, N., Takagi, H.: Visualized iec: interactive evolutionary computation with multidimensional data visualization. In: IECON 2000. 26th Annual Conference of the IEEE. Volume 4. (2000) 2738 – 2743

[20] Llorà, X., Sastry, K., Alías, F., Goldberg, D.E., Welge, M.: Analyzing active interactive genetic algorithms using visual analytics. In: Proceedings of the 8th annual conference on Genetic and evolutionary computation. GECCO '06, New York, NY, USA, ACM (2006) 1417–1418

[21] Elmqvist, N., Dragicevic, P., Fekete, J.D.: Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis 2008) **14**(6) (2008) 1141–1148

[22] Bezerianos, A., Chevalier, F., Dragicevic, P., Elmqvist, N., Fekete, J.D.: Graphdice: A system for exploring multivariate social networks. Computer Graphics Forum (Proc. EuroVis 2010) **29**(3) (2010) 863–872

[23] Bezerianos, A., Dragicevic, P., Fekete, J.D., Bae, J., Watson, B.: GeneaQuilts: A System for Exploring Large Genealogies. IEEE Transactions on Visualization and Computer Graphics **16** (10 2011) 1073–1081 Genealogy visualization, interaction.