Visualization of Evolutionary Algorithms - Set of Standard Techniques and Multidimensional Visualization

Hartmut Pohlheim
DaimlerChrysler AG, Research and Technology
Alt-Moabit 96a, 10559 Berlin, Germany
pohlheim@dbag.bln.daimlerbenz.com

Abstract

Evolutionary algorithms work in an algorithmically simple manner but produce a vast amount of data. The extraction of useful information to gain further insight into state and course of the algorithm is a non-trivial task. In this paper we present a set of standard visualization techniques for different data types and time frames of the evolutionary algorithm. The methods were selected according to their usefulness for real world applications, and tested during the solution of some complex real world optimization problems. Additionally, multidimensional scaling as a technique for the presentation of high-dimensional data with standard visualization techniques is presented. We demonstrate the use of this technique for the visualization of the "path through the search space" of the best individuals during an optimization run, and for the comparison of multiple runs regarding the variables of individuals and multi criteria objective values ("path through solution space").

1 INTRODUCTION

Evolutionary algorithms work in an algorithmically simple manner. When put to work they produce a vast amount of data. Apart from simple convergence information it is a non-trivial task to extract useful information from those data to provide insight into the state and progress of the evolutionary algorithm. Methods for extracting and visualizing relevant data are still under development ([9], [10]).

In this paper a set of standard visualization techniques for different data types is presented. These data types and techniques give an instant visual impression of the evolutionary algorithm's progress and the actual state of the individuals of the population. The methods were selected according to their usefulness for real world applications. Hopefully, the defined standard set will open up a discussion between users of evolutionary algorithms about necessary and useful visualization techniques.

Beside the standard set, an advanced technique for visualizing multidimensional data produced by the evolutionary algorithm is presented - multidimensional scaling. This advanced technique is independent of the representation of the variables. The technique is applied to the visualization of variables of individuals of a population, and demonstrated for the visualization of multi criteria objective values in multiobjective optimization. This technique is especially useful for the comparison of high-dimensional data of different runs.

In Section 2 a standard set of visualization techniques is proposed. Section 3 describes multidimensional visualization and presents some examples. Section 4 gives concluding remarks and some prospect directions of further investigations.

2 SET OF STANDARD VISUALIZATION TECHNIQUES

In this chapter a number of data types and techniques useful for visualizing the state of the population and the progress of the evolutionary algorithm are proposed. These techniques are simple to calculate and easy to display using standard visualization tools. This set of techniques provides a base line for understanding the evolutionary process.

An overview of the different data types and time frames useful for the visualization of evolutionary algorithms is given in figure 1. For further description, the techniques are systematized into two categories, depending on the time range represented by the data. The first category encloses methods for visualizing data produced during many generations, thus presenting a picture of the progress or course of the evolutionary algorithm. The second category contains techniques to visualize data produced anew for every generation. These techniques give a picture of the current state of the population or the evolutionary algo-

- individuals (solution vector)
  - → variables
  - → objective value(s)
- (sub) population
  - → individuals (variables/objective values of best/all individuals)
  - → distance between individuals
  - → ranking / order and size of subpopulation

◆ different time frame

- → one generation ➡ state of EA
- → multiple / all generations ➡ course of EA
- → multiple runs ➡ comparison of EA

Figure 1: Data Types and Time Frames of Evolutionary Algorithms for Visualization

rithm. Examples for the visualization of multiple runs (comparison of different EA) are given in Section 3.

For each data type proposed a number of techniques are specified and visual examples are given. The information to be extracted from the respective graphics is briefly described.

2.1 VISUALIZATION OF THE COURSE OF THE EVOLUTIONARY ALGORITHM

The following data types are used for visualizing the progress of the evolutionary algorithm:

1. objective value of best individual of every generation (convergence diagram),
2. variables of best individual of every generation,
3. objective values of all individuals of all generations.

2.1.1 Best Objective Value (Convergence Diagram)

The visualization of the objective value of the best individual of every generation is used very often, figure 2 left. Depending on the scaling of the objective values, this diagram may give a good impression of the convergence of the evolutionary algorithm.

Many publications also include the mean and worst objective value in the convergence diagram. However, including the worst objective value very often destroys the scaling of the diagram. More information can be extracted, if the mean objective value and the standard derivation of all objective values are included instead, figure 2 middle. This proved appropriate for many real world problems solved using evolutionary algorithms.

When the regional model (often called migration model or coarse grained model) is employed, the simple convergence plot can be extended to show the best objective value of each subpopulation, figure 2 right. This is of further advantage when different strategies for every subpopulation are used.

2.1.2 Variables of Best Individual

The visualization of the variables of the best individual of every generation is one of the most representative diagrams for the progress of an optimization run. The user can actually see how the variables are changed and which combinations of variables are successful. During the run, the user gets a picture of good variable values.

Figure 3 presents three different techniques for this data type. The simplest one is the 2-D line plot, figure 3 left. Every line represents the course of one variable. However, the assignment between variable number and line is not possible (without a legend). The 3-D line plot, figure 3 middle, provides the extra dimension needed to assign every variable visually to a line. Nevertheless, the best

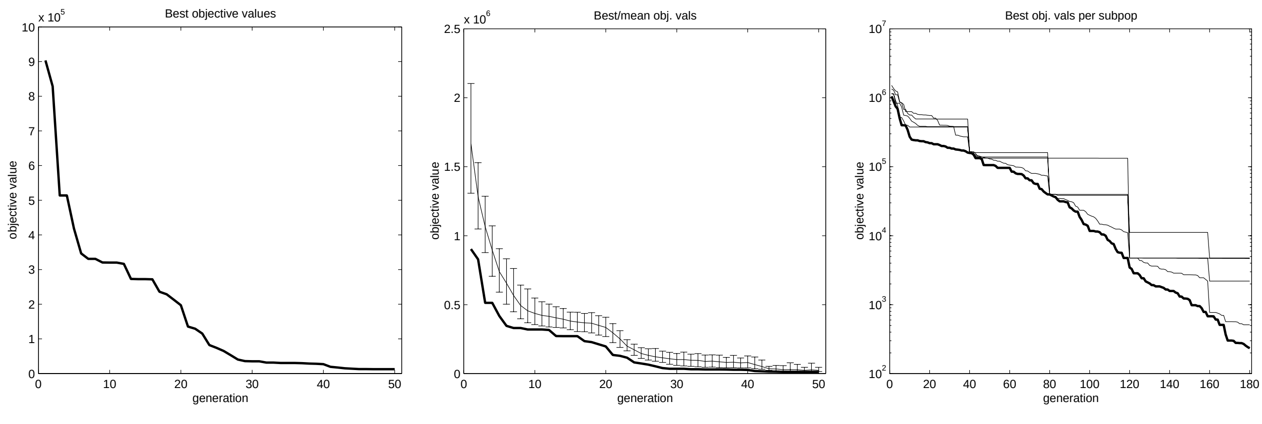

Figure 2: Convergence Diagram; left: Objective Value of Best Individual, middle: Best and Mean Objective Value including Standard Derivation of All Objective Values, right: Best Objective Value of Every Subpopulation

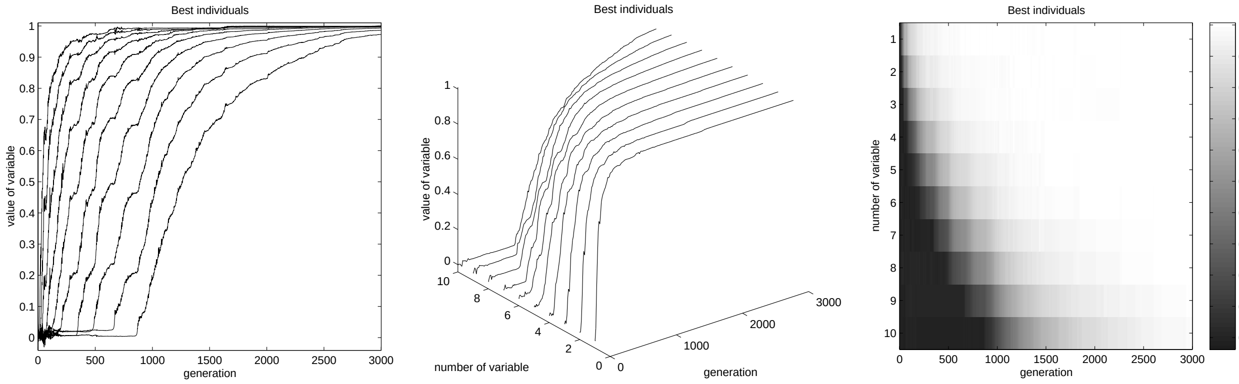

Figure 3: Variables of Best Individual of Many Generations; left: 2-D Line Plot; middle: 3-D Line Plot, right: 2-D Image Plot

technique to extract all this information is a 2-D image plot, figure 3 right. Even for a large number of variables and different values of adjacent variables, a good impression of the distribution and change of the variables is possible.

Very different ranges of the variables can be circumvented by scaling the variables according to the domain of each variable, thus visualizing the relative values of the variables.

2.1.3 Objective Values of All Individuals

The previous two data types involved just the best individual of the population. To get an overview of the whole population, the objective value of all individuals of every generation can be visualized.

Two techniques are demonstrated. The 2-D point plot, figure 4 left, shows the distribution of the objective values. The 2-D image plot, figure 4 middle and right, additionally shows the assignment between objective value and position of the individual in the population. This is especially useful when the regional model, figure 4 middle, or the local model, figure 4 right, is employed. In figure 4 middle the changing size of the subpopulations due to competing subpopulations is apparent.

2.2 VISUALIZATION OF THE STATE OF THE EVOLUTIONARY ALGORITHM

The following data types and techniques are proposed for visualizing the actual state of the evolutionary algorithm:

1. objective value of all individuals,
2. variables of all individuals,
3. distance map of objective values and distance map of variable values.

2.2.1 Objective Values of All Individuals

Similar to the variables of the individuals, the objective values of all individuals of one generation give a good impression of the state of the population.

Figure 5 presents three techniques for visualizing the objective values. The 2-D point plot in figure 5 left is straightforward. By using a filled 2-D stairs plot, figure 5 middle, the differences inside the population and adjacent individuals are more apparent. Both diagrams display a population divided into subpopulations, the dotted lines

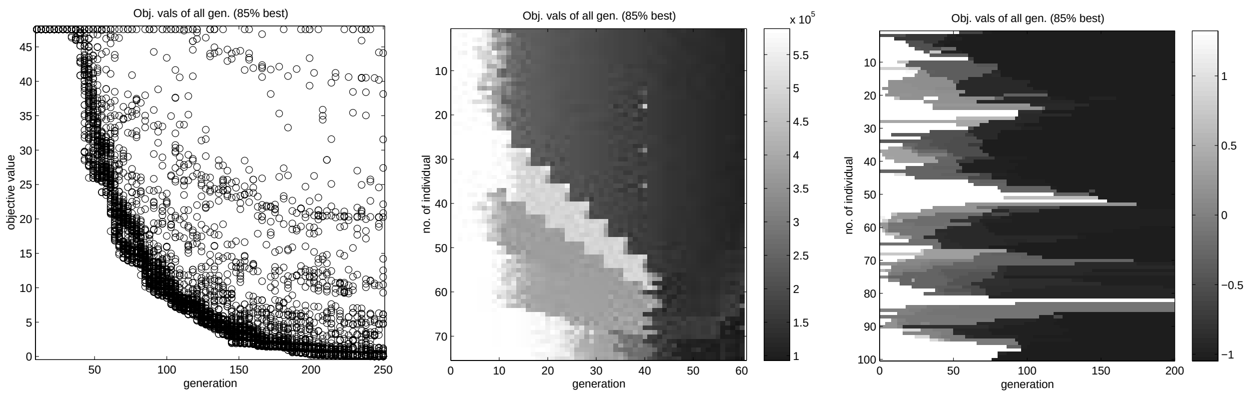

Figure 4: Objective Values of All Individuals of Many Generations; left: 2-D Point Plot, middle: 2-D Image Plot employing a Regional Model with Competing Subpopulations, right: 2-D Image Plot employing a One-Dimensional Local Model

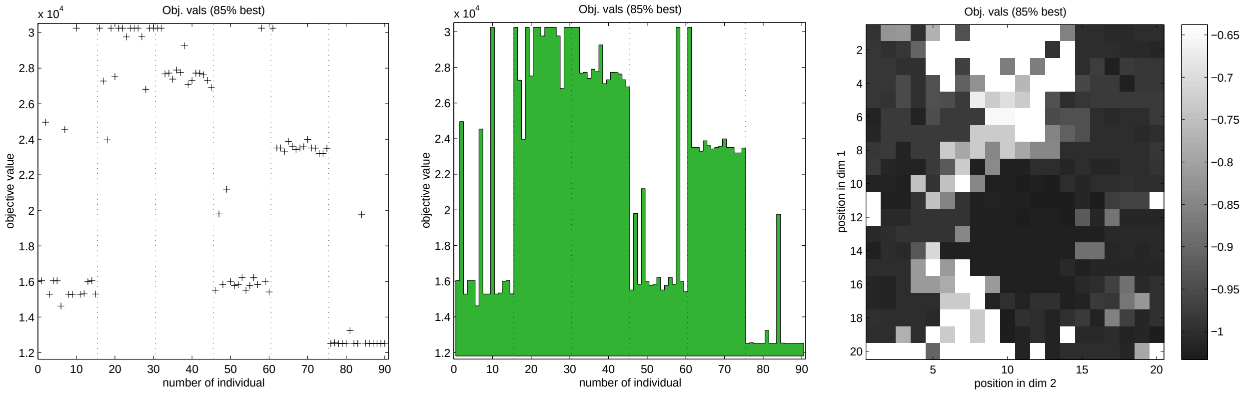

Figure 5: Objective Values of All Individuals of a Population in One Generation; left: 2-D Point Plot, middle: 2-D Stairs Plot, right: 2-D Image Plot employing a Two-Dimensional Local Model

display this division. Figure 5 right employs a 2-D image plot to visualize the two-dimensional neighbourhood of the objective values when using a two-dimensional local model. A sequence of such diagrams shows the diffusion of good objective values through the population during an optimization run.

2.2.2 Variables of All Individuals

The visualization of the variables of the individuals of a population provides an insight into the distribution of the individuals, the distance between individuals and the concentration of the individuals in one or more areas. However, because of the enormous amount of information the chosen technique for visualization is important.

Figure 6 shows one technique at three different stages of an optimization run. A 2-D line plot of the variables of the best individual together with the mean and the standard derivation of the variables is used. The sequence of three diagrams displays the change of the variables values and their distribution.

2.2.3 Distance between Individuals

All previous data types were directly produced by the evolutionary algorithm. The distance between individuals, however, is a derived data type. For the calculation of the distance, the Euclidean distance between the variables of the individuals is used.

If there is no neighbourhood relationship between individuals (global and regional model), the distance distribution is visualized using a 2-D histogram stairs plot, figure 7 left. This diagram directly shows the distribution of the individuals.

When the local model is used a neighbourhood relationship is defined. Thus, the distance should be visualized using the individuals in their respective neighbourhood. In a one-dimensional local model every individual has two neighbours, in a two-dimensional local model four neighbours. The distance between individuals of a one-dimensional local model is shown in figure 7 middle, employing a 2-D stairs plot. The 2-D image plot in figure 7

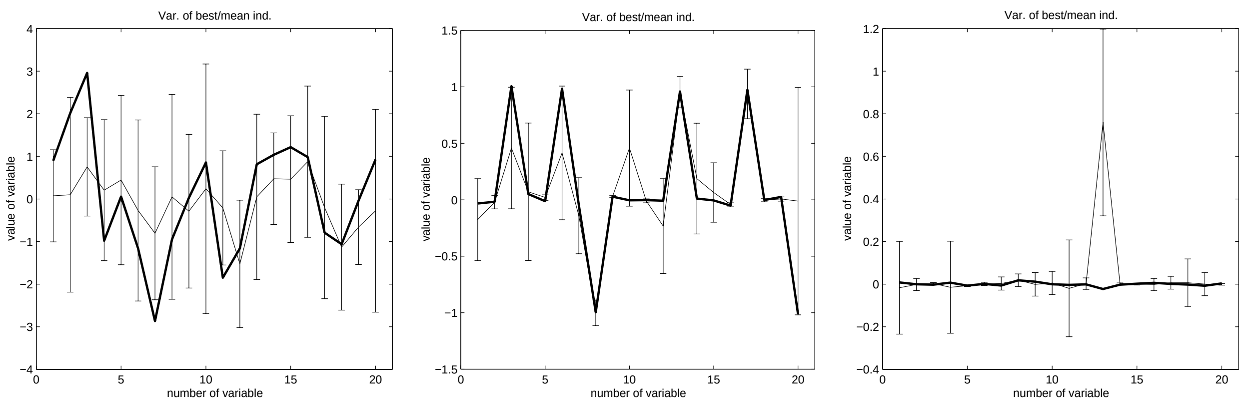

Figure 6: Variables of All Individuals of a Population: 2-D Line Plot of the Best (thick line) and Mean Individual including Standard Derivation of All Variables at Three Stages of an Optimization Run; left: Beginning of Run, middle: Middle of Run, right: End of Run

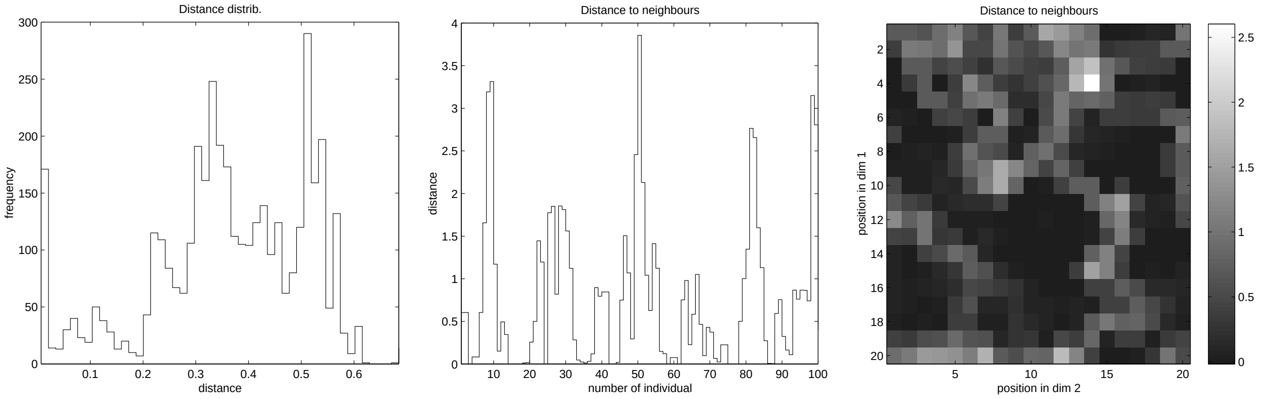

Figure 7: Distance between Individuals of a Population; left: 2-D Stairs Plot of Distance Distribution between all Individuals, middle: 2-D Stairs Plot of Distance Map (Distance between Neighbours) employing a One-Dimensional Local Model, right: 2-D Image Plot of Distance Map (Distance between Neighbours) employing a Two-Dimensional Local Model

right displays the distance between individuals in a two-dimensional local model.

All data types and techniques visualizing the state of a population or the evolutionary algorithm should be used in a sequence of the respective diagrams during an optimization run or afterwards. The animation of these diagrams can reveal new and extended information and insights into the problem at hand.

2.3 GRAPHICAL USER INTERFACE

We developed a graphical user interface to access all the different visualization methods and styles during and after an optimization (a text based user interface was also used earlier, but is not as intuitive as a graphical user interface). The GUI is part of the Genetic and Evolutionary Algorithm Toolbox for Matlab - GEATbx [5].

Data needed for visualization are collected during an optimization run. The data are saved in a binary file, thus making them accessible later. This offers possibilities for using the same program and data files for online and off-line visualization.

The user interface allows the selection of single and multiple visualization methods. The user can select which time frame of the optimization should be shown (how many generations for visualization methods displaying the course of an optimization and/or the generation number for visualization methods showing the state of an optimization). For each method, any of the available styles can be selected.

Figure 8 shows the graphical user interface in off-line mode. Six visualization methods are selected, just one (distance of individuals) is not used here. For each method, one style is chosen. In addition to the method and style selection, the time frame (beginning and end generation of the visualized data) can be adjusted in the range of available values. For a more comfortable viewing of the results special buttons are included to go forward (advance time frame for 5 generations) or back in the visualized time frame.

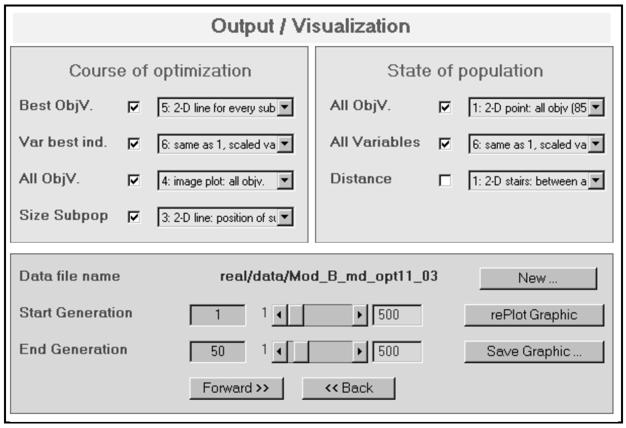

Figure 8: The Graphical User Interface corresponding to the Visualization Example in Figure 9

The result of this configuration used for the visualization of the result of an optimization run (optimization of the maximal run time of a real time software module employed for the control of specific aspects in a vehicle system) is shown in figure 9. The 3 diagrams at the top represent the course of the optimization for the first 50 generations of the best objective values, the variables of the best individuals and the objective values of all individuals. The two diagrams, bottom left and middle, show the state of the population in the 50^{th} generation, the variables of all individuals and the objective values off all individuals. The last diagram displays the ranking of the subpopulations during the optimization run and can be used to access the performance of the multiple strategies used (each subpopulation used a different strategy, i.e., different search parameters).

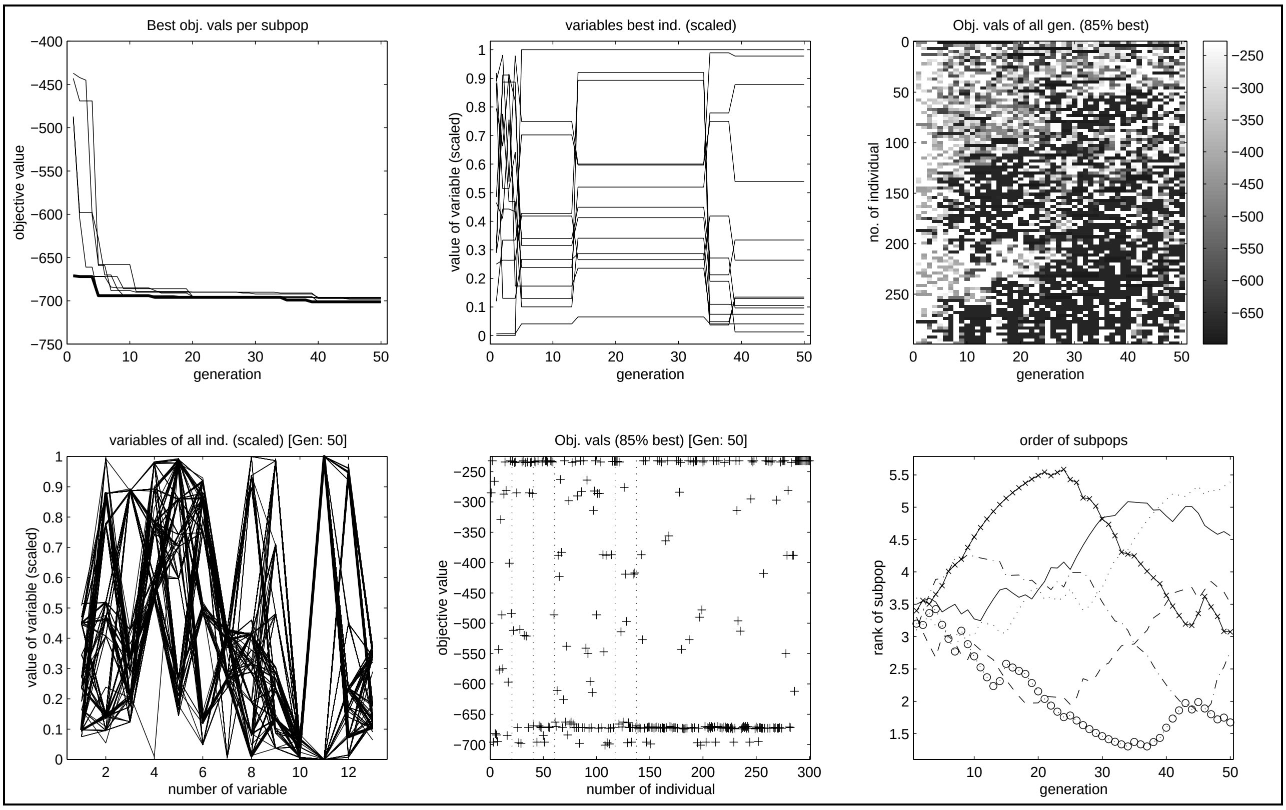

Figure 9: Example of the use of some visualization methods in one figure (optimization of real time system for maximal run time, all objective values are multiplied with -1 to accommodate a minimization)

The flexibility offered by the user interface gives the user a powerful tool, easy to use for the thorough examination and for the detailed documentation of various aspects of the optimization runs.

3 MULTIDIMENSIONAL VISUALIZATION

Most of the commonly used techniques for visualization are limited to representing data depending on one or two variables. This is due to the human visual limitation to three dimensions. There are two possible extensions to go beyond this limitation: using color for the fourth dimension and time as the fifth dimension. Neither possibility is very common and requires practice, especially if time is used for visualizing the fifth dimension. However, if the problem incorporates more than five dimensions a new method for visualizing arbitrarily high dimensions must be found.

For the visualization of multidimensional data a method to transform multidimensional data to a lower dimension is needed, preferably to 2 or 3 dimensions. This transformation should provide a lower-dimensional picture where the dissimilarities between the data points of the multidimensional domain corresponds with the dissimilarities of the lower-dimensional domain. These transformation methods are referred to as ‘multidimensional scaling’ ([2], [8]).

To measure the dissimilarity, the distance between pairs of data points is used. These distances can be genuine distances in the respective high-dimensional domain, for instance the Euclidean distance. If a genuine distance measure is not applicable, the dissimilarities can be defined by a substitute measure. The distances need not satisfy the triangle inequality ($d_{ik} \le d_{ij} + d_{jk}$, d: distance between data points) thus the term dissimilarity is used. In [8] examples of non-metric measures of categorical data are given (for instance, simple matching coefficient).

3.1 CALCULATION OF LOW-DIMENSIONAL REPRESENTATION

For finding a low-dimensional representation, y_i, of a number of points in a high-dimensional domain, x_i, the distance between all points in their respective domains is calculated.

$x_1,\ldots,x_n$: data points in $\Re^p$

$y_1,\ldots,y_n$: data points in $\Re^q$, $p \ge q$

δ_{ij}: distance between x_i und x_j

d_{ij}: distance between y_i und y_j

Then a configuration of low-dimensional image points, y_i, is looked for, such that the distances d_{ij} between image points are as close as possible to the corresponding original distances δ_{ij}. Since it will usually not be possible to find a configuration for which $d_{ij} = \delta_{ij}$ for all i and j, a quality criterion, J_{ef}, for ranking different configurations must be defined, equation 1.

$$J_{ef} = \frac{1}{\sum\limits_{i<j} \delta_{ij}} \sum_{i<j} \frac{\left(d_{ij} - \delta_{ij}\right)^2}{\delta_{ij}} \tag{1}$$

An optimal configuration minimizes the criterion function and can be sought by a gradient descent method. The gradient of the criterion function is easy to compute, equation 2.

$$\nabla_{y_k} J_{ef} = \frac{2}{\sum\limits_{i<j} \delta_{ij}} \sum_{j \neq k} \left(\frac{d_{kj} - \delta_{kj}}{\delta_{kj}} \cdot \frac{y_k - y_j}{d_{kj}} \right) \tag{2}$$

One of the best known methods for multidimensional scaling is SAMMON mapping [11]. SAMMON used a Newton method (steepest decent). However, this method is not very robust and diverges without special interaction [8].

For optimization we used two different methods. First, a standard optimization method included in Matlab was employed (BFGS Quasi-Newton method with a mixed quadratic and cubic line search procedure - [4]). Later, a more robust search method was employed, the RPROP algorithm [7]. The RPROP algorithm uses just the change of the sign of the gradient for step size control. RPROP is widely used in the field of neural networks. Both algorithms produced good results. However, when using the Quasi-Newton method, multiple runs must be performed and the received results must be valued by the user. The RPROP algorithm ran slower but produced more consistent results.

The starting configuration can be chosen randomly. However, using a principal component analysis for defining the starting configuration, the convergence of the optimization can be accelerated. Nevertheless, because of the local search the optimization often gets stuck in local optima, thus multiple runs using different initializations were performed. After comparing the results the best configuration was chosen.

Several papers on the use of multidimensional scaling for the visualization of evolutionary algorithm have been published recently ([1], [3] and [12]).

3.2 VISUALIZATION EXAMPLES OF MULTIDIMENSIONAL SCALING

This section presents visualization examples that use multidimensional scaling to map data to a low-dimensional domain for visualization.

- visualization examples of individuals of a population,
- visualization examples of non-dominated solutions in multiobjective optimization.

3.2.1 Variables of Best Individual of All Generations

This example shows the „path through search space“ of the best individual of every generation. The resulting diagram produces a clearer picture than visualizing the best individual of all generations. By multidimensional scaling the variables of the best individuals of all generations were reduced to two dimensions. These low-dimensional data were visualized employing a 2-D point plot. The label of every point is the generation of the respective individual.

Figure 10 presents the results of two different objective functions. The left plot is a run of ROSENBROCKs function, often called ‘banana function’. The used function employed 10 dimensions. The optimization ran over 3000 generations. The plot clearly shows the curve-shaped characteristic of this function known from 3-D variation plots of two variables for this function.

The second example, figure 10 right, shows the result of a real world application, the optimization of a Chopper (DC/DC line-side converter), where one individual consisted of 9 variables. The characteristic of the path of the best individual through search space is quite different compared to ROSENBROCKs function. During the first 5 generations the best individuals are in one area. In generation 10 the best individual is quite far away. Through all further generations the best individuals can be found in a very small area.

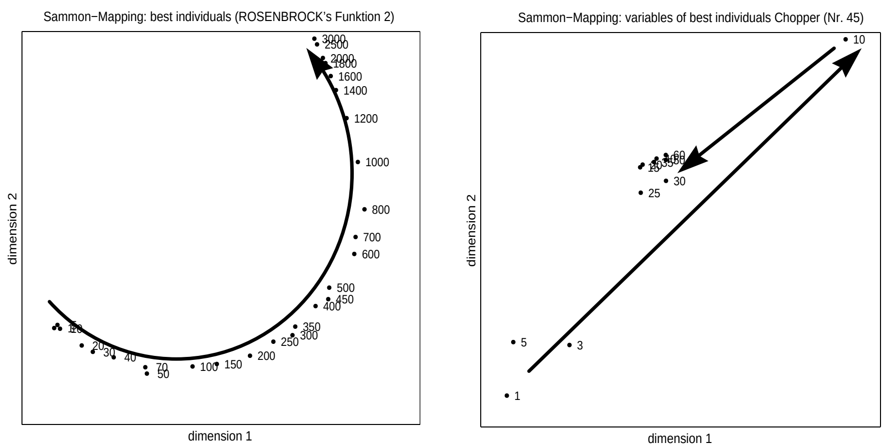

Figure 10: Multidimensional Scaling of Variables of Best Individuals of One Run; left: ROSENBROCKs Function, right: Chopper System (real world example)

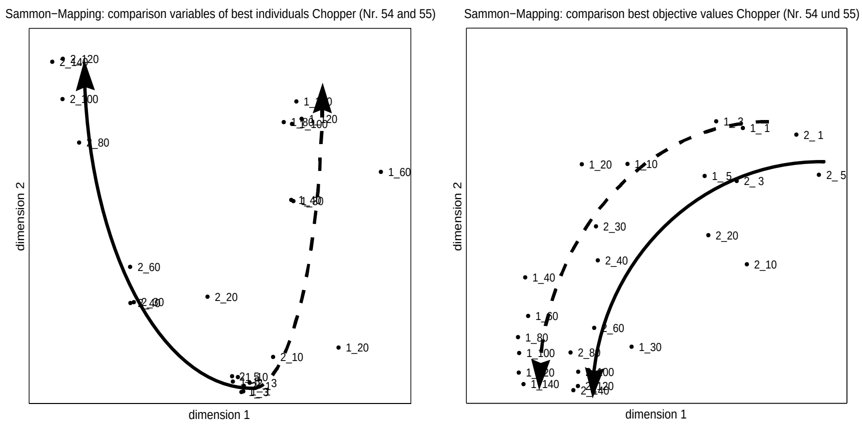

Figure 11: Comparison of Two Optimization Runs of the Chopper Application using Multidimensional Scaling; left: Multi Criteria Objective Values, right: Variables of Corresponding Individuals

3.2.2 Multiple Objective Values of Best Individual of Every Generation

The second example presents the multidimensional scaling of multi criteria objective values. Again we use the real world Chopper example. The objective function consisted of the simulation of up to nine different scenarios, where each scenario produced one objective value. The sum of all separate objective values was used as the objective value for the evolutionary algorithm. However, it proved difficult to compare the course of two different runs, but was solved using multidimensional scaling.

Figure 11 shows the comparison of two optimization runs of the Chopper. On the left side the visualization of the multi criteria objective values is shown, on the right the best individuals of each generation. The label of each point consists of the number of the run followed by the number of the generation. The diagrams show that both runs start from the same area of the objective space or variable search space. However, during the run the objective values and the individuals both diverge to different areas along a line each. Without multidimensional scaling it would be very difficult to extract this information from the results.

4 CONCLUDING REMARKS

This paper proposed multidimensional scaling for visualization of data produced by evolutionary algorithms. The examples presented showed the advantage of this technique. Information which is not available or very difficult to extract using other techniques can be derived. Multidimensional scaling opens a field for new visualization techniques not only applicable to the domain of evolutionary algorithms.

The proposed set of "standard" visualization techniques of evolutionary algorithms provides a baseline for understanding the evolutionary process. Combined with advanced techniques, powerful visualization tools to aid the designer and user of evolutionary algorithms can be constructed.

The presented visualization techniques were applied in the course of solving real world problems using evolutionary algorithms. Thus, these techniques are not just proposed, but have been used and proved informative (see [6]). An example implementation of all these and additional techniques can be found in [5].

References

[1] *Collins, T. D.*: Genotypic-Space Mapping: Population Visualization for Genetic Algorithms. Technical Report, KMI-TR-39, Knowledge Media Institute, The Open University, Milton Keynes, UK, 1997.

[2] *Cox, T. F. and Cox, M. A. A.*: Multidimensional Scaling. London: Chapman & Hall, 1994.

[3] *Dybowski, R., Collins, T. D. and Weller, P. D.*: Visualization of Binary String Convergence by Sammon Mapping. in *Fogel, D. B.*: Evolutionary Programming V, Proceedings of the Fifth Annual Conference on Evolutionary Programming. Cambridge, Massachusetts: MIT Press, 1996.

[4] *Mathworks, The*: Matlab - UserGuide. Natick, Mass.: The Mathworks, Inc., 1994-1999. http://www.mathworks.com/

[5] *Pohlheim, H.*: Genetic and Evolutionary Algorithm Toolbox for Matlab. http://www.geatbx.com/, 1994-1999.

[6] *Pohlheim, H.*: Entwicklung und systemtechnische Anwendung Evolutionärer Algorithmen. Aachen, Germany: Shaker Verlag, 1998. (Development and Engineering Application of Evolutionary Algorithms. in german)

[7] *Riedmiller, M. and Braun, H.*: A direct adaptive method for faster backpropagation learning: The RPROP algorithm. in H. Ruspini (editor), Proceedings of the IEEE International Conference on Neural Networks (ICNN), pp.586-591, 1993.

[8] *Ripley, B. D.*: Pattern Recognition and Neural Networks. Cambridge, GB: Cambridge University Press, 1996.

[9] *Routen, T. W. and Collins, T. D.*: The Visualisation of AI Techniques. Proceedings of Third International Conference on Computational Graphics and Visualisation Techniques COMPUGRAPH'93, Alvor, Portugal, New York, USA: ACM Press, pp. 274-282, 1993

[10] *Routen, T. W.*: Techniques for the Visualisation of Genetic Algorithms. in Proceedings of The First IEEE Conference on Evolutionary Computation, Piscataway, New Jersey, USA: IEEE Service Center, Vol. II, pp. 846-851, 1994.

[11] *Sammon, J. W. jr.*: A Nonlinear Mapping for Data Structure Analysis. IEEE Transactions on Computers, vol. C-18, no. 5, pp. 401-409, 1969.

[12] *Shine, W. and Eick, C.*: Visualizing the evolution of genetic algorithm search processes. In The Proceedings of the 1997 IEEE International Conference on Evolutionary Computation ICEC'97, pp. 367-372, 1997.