Utilizing Gradient Analysis within Interactive Genetic Algorithms

Matthias Kulcke[1], Wolfgang E. Lorenz[2]
[1]Hamburg University of Technology/HafenCity University Hamburg [2]Vienna University of Technology
[1]matthias.kulcke@tu-harburg.de [2]lorenz@iemar.tuwien.ac.at

The paper describes and discusses the possible integration of gradient analysis, as a method and tool for architects and designers to analyze the degree of proportion-complexity of a design, into the process of designing an object utilizing interactive genetic algorithms (IGA). A VBA implementation for AutoCAD has been developed by the authors, enabling to test the usability of genetic algorithms (GA) for minimizing the angle-redundancy and length-redundancy quotient. The gradient analysis itself has been developed on the basic assumption that the complexity of an objects appearance is reduced by redundancy, which can be measured focussing on different levels of comparison; among others e. g. variety of material, colour-combinations and proportion. The latter comes under scrutiny if the method of gradient analysis is applied.

Keywords: *Gradient Analysis, Interactive Genetic Algorithm, Design Complexity, Redundancy, Spatial Analysis, Form and Geometry, Proportion*

THE ALGORITHM

The complexity of a given architectural object or object arrangement can be measured on the level of gradient relations, the gradients itself being the representations of the proportion of each rectangle that is defined by each pair of geometric points. Using the gradient analysis means regarding the frequency of repetition of gradients, comparing every significant point with all the other points in a 2D CAD representation of the object one by one and successively listing pairs of points and their gradients (Kulcke and Lorenz 2015). In addition the distance between points is taken into account; the gradient analysis returns an angle quotient and a length quotient, each of which can be used as a fitness-value within a genetic algorithm. This is tested by the authors on simple facade elements with fixed main outlines and two openings with varying corner-points. The coordinates of the corner-points are translated into binary DNA (1st step: encoding). After using crossover (2nd step; with or without fitness check beforehand) and mutation (3rd step) on a population of n-parents (chromosomes), angle quotient and length quotient are used as fitness-values (4th step) to determine the segment-size on Goldbergs weighted roulette wheel (see Coates 2010, König 2010). By digitally spinning the roulette wheel, members for the succeeding breeding population are picked to be proposed (element of chance); out of these the user choses 1 or 2 favourites. Taking this choice into account the next breeding population goes through the algorithm (see flowchart, Figure 1).

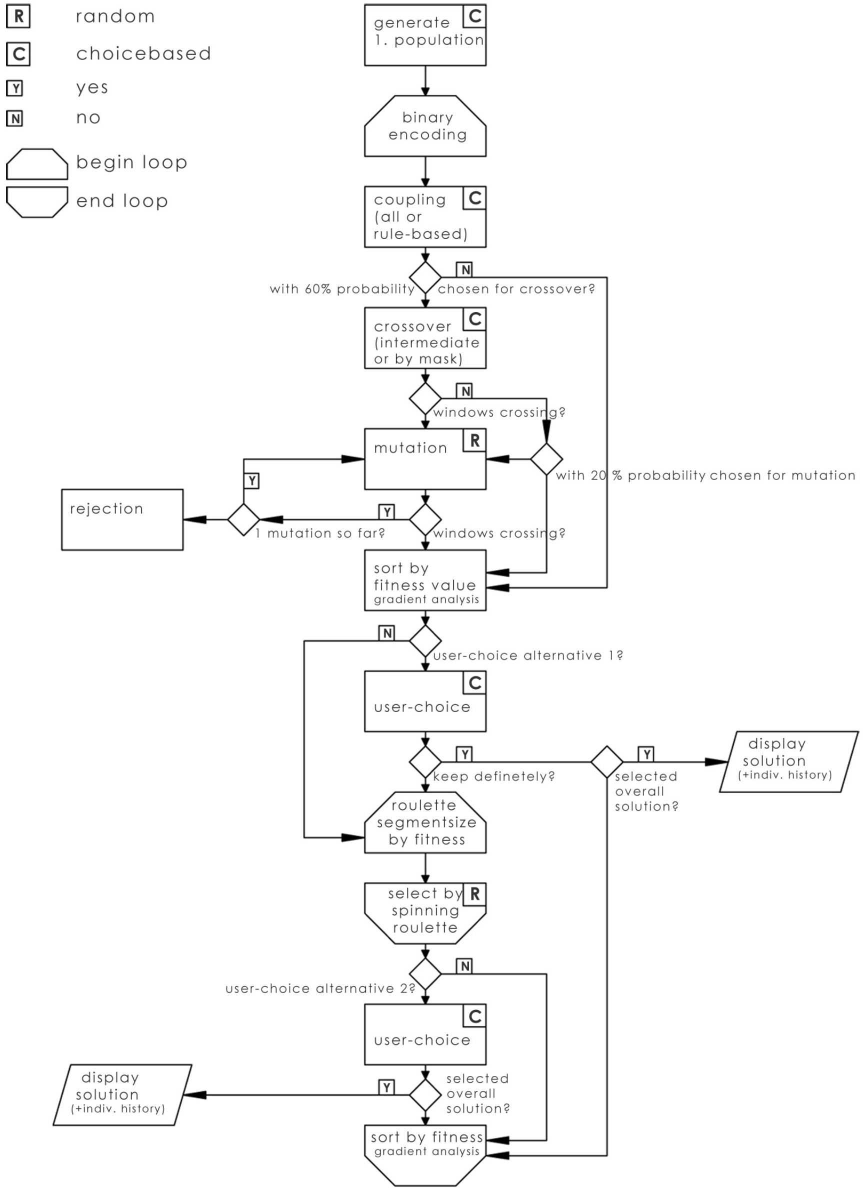

Figure 1
Flowchart representing the interactive genetic algorithm.

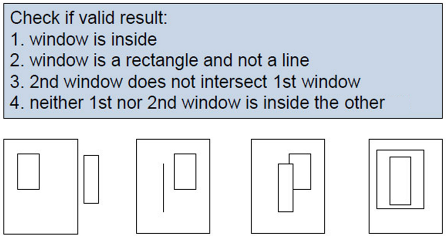

Figure 2
Invalid results.

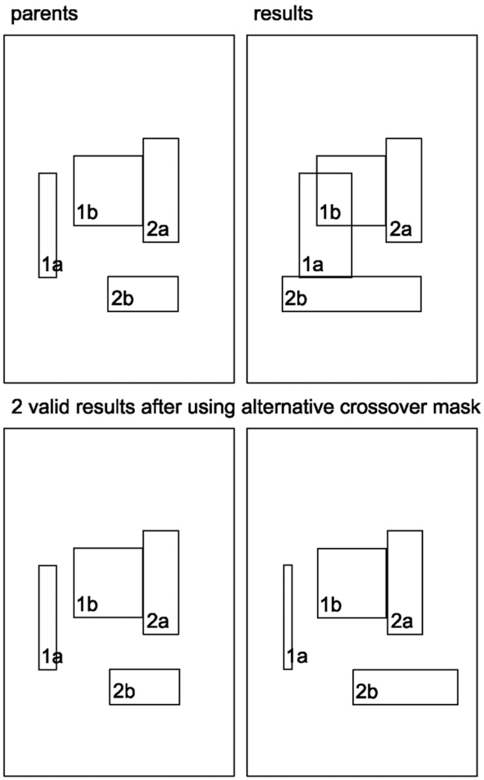

Figure 3
Using an alternative crossover mask to achieve valid results.

Alternative Use of the Roulette Wheel

Different orders of the steps for reaching the next population are also possible. Poirson et al. have proposed the application of an interactive genetic algorithm in design tasks, that allows users to influence the choice of candidates out of each succeeding population (Poirson et al. 2010). They have applied this interactive genetic algorithm e. g. integrating customers in the process of a car dashboard design by giving them an automatically generated selection of visualisations of individuals of each generation to chose 1 or 2 favourites from. In their case the user's choice influences the segment size of the chosen individual on the roulette wheel which is spun afterwards.

Using Different Crossover Masks Depending on the Results

The algorithm applies crossover and mutation in the automated first part. If the crossover of two parents fails to produce a valid child, e. g. a set of coordinates that creates two window openings with overlap or other flaws (Figure 2), then the binary description of these coordinates is mutated. Another approach could be to use a different mask for the crossover if invalid solutions are produced (Figure 3).

THE AUTOMATED PART OF THE ALGORITHM

Steps 1 to 4 of the algorithm (for the time being still limited to a fixed rectangle, defining the main outline and two openings) make up the automated part without possibility for user intervention. Both, step 2 (crossover) and step 3 (mutation) are for the time being applied to each single element (of the pool of possibilities) influenced by probability values. Crossover masks have been applied in two ways: either the binary code is cut into two pieces at any position or just in the middle. The latter results in simply switching the coordinates for the second window, since the program currently takes only two windows into account (each consisting of 4 values: the x- and y-coordinates of the bottom left corner and the x- and y-coordinates of the top right corner). The authors utilized different versions of the program to adjust the automated part of the IGA with adjustments

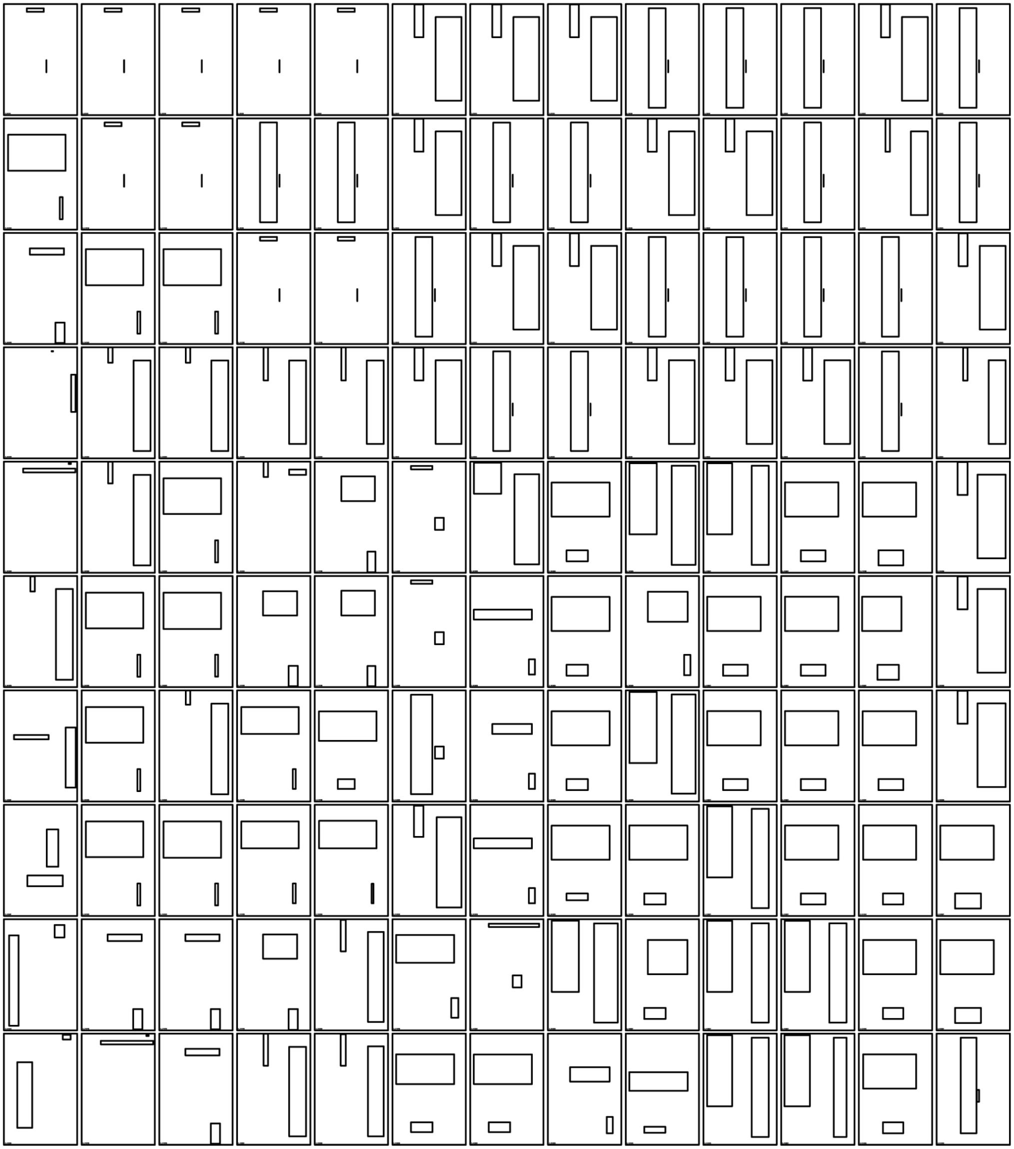

Figure 4
Development of the facade through 12 generations.

still ongoing. The macro program used in AutoCAD returns a set of designs starting with the initial individuals in the left column, with fitness decreasing from top to bottom row. From left to right different time steps are visualised with a finishing result on the right. Differences between elements of the last column are minimised as a (local) optimum is reached (Figure 4).The initial aim to reach a design that produces a minimal angle-redundancy quotient and a minimal length-redundancy quotient is questioned by the results. Left unchecked by rules and constraints this aim produces openings decreasing in size, which is the logical consequence if the only goal is to escape complexity in proportion as much as possible (Figure 5).

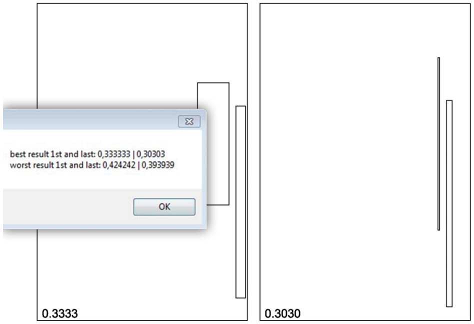

Figure 5
Openings decreasing in size/tending toward extreme proportions.

THE FITNESS FUNCTION AND USER CHOICE INTEGRATION

The integration of gradient analysis in an interactive genetic algorithm aims rather at didactic and professional application as opposed to customer-use, therefore it should be possible to optionally add the fitness-values of each individual of the population the user can choose from in text. This is done in a two step process; first, the favourites are chosen by visual preference, than the fitness-value representing the degree of complexity is made to appear and the user can alter his decision, which is now influenced by intuition, perception and analysis-output. To allow for comfortable interactivity and easy access the algorithm is rewritten in PHP and using the <canvas> element in HTML5 (Figure 6).

Left uninfluenced in the individual evolutionary steps, the genetic algorithm aiming for the lowest angle-quotient possible applying the gradient analysis as a fitness value shows certain tendencies that might not be wished for in a real building environment, e.g. window/door-openings tend to go for extreme i.e. slim proportions resp. decrease in the size of their area. In a fully automated context this would call for corrective rules and constraints, in the case of an interactive genetic algorithm avoiding extreme proportions may be left to the user, if he/she wishes to avoid them. In addition the degree of complexity reduction could also be chosen by the user adjusting the angle-quotient; an ideal angle-quotient could be set by the user, either beforehand or in the course of searching for the facade wished for. Functions could be added accordingly to the proposed IGA.

DISCUSSION AND OUTLOOK

The current results are promising as the fitness of the results increases after only a few steps. However, further program adjustments have to be be examined in detail. Designs allowing for more windows will be tested as well, in order to gain experience regarding the performance. The integration of user choices is of main focus as the authors intend the tool to be utilized as a cognitive and analytic aid during the design process and not as an automatic design generator. Currently the determination of the best position of user intervention within the GA, thus making it interactive, is still ongoing. An advantage of using interactive genetic algorithms instead of fully automated ones is that fewer iterations are needed to come to a conclusion, but more importantly a dialogue between the algorithm and the user is introduced. With the integration of Gestalt related methods like the gradient analysis, interactive genetic algorithms become didactic instruments on the aesthetic and cognitive level, combining theory and design practise.

To further develop an asynchrone dialogue between designers and users in a form that conse-

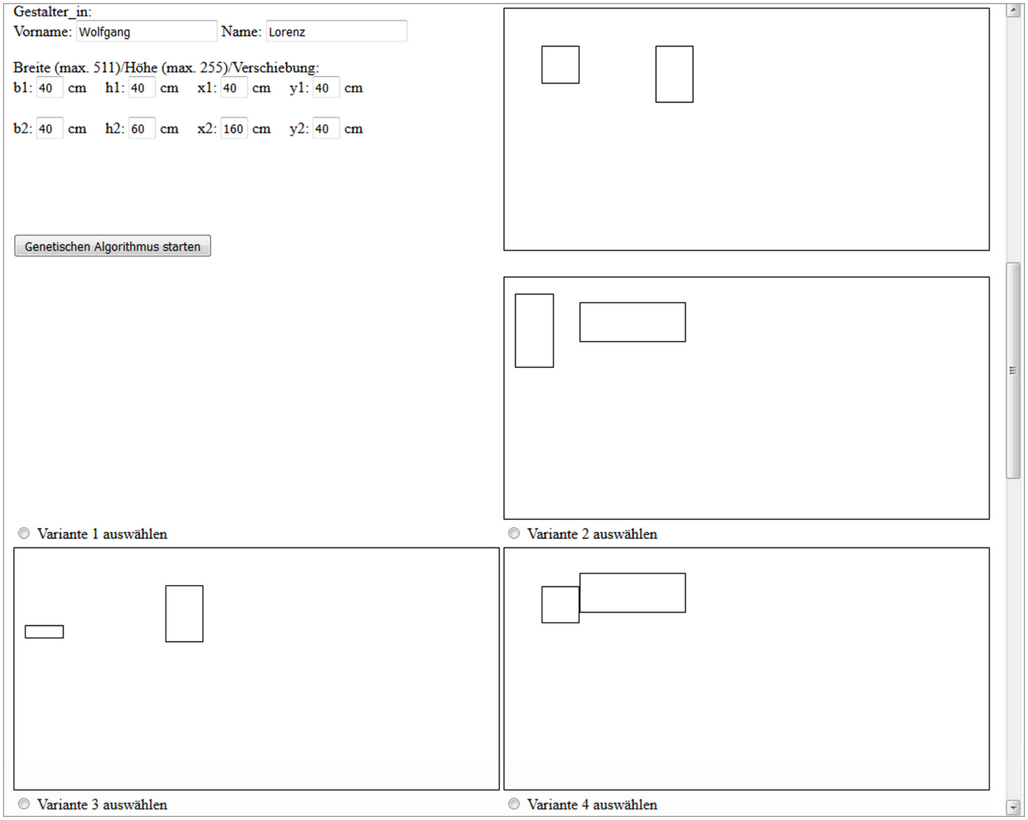

Figure 6
The algorithm in PHP and HTML5.

quently upholds the initial idea of user-involvement through an IGA that still allows for professionally responsible solutions, a next step will be to list and formulate necessary rules and constraints that can be used for an accompanying or closing analysis of the (final) instances that are created by the genetic algorithm in collaboration with the users' choices.

REFERENCES

Coates, P 2010, *The Programming of Architecture*, Routledge, New York

Kulcke, M and Lorenz, W 2015 'Gradient-Analysis: Method and Software to Compare Different Degrees of Complexity in the Design of Architecture and Designobjects', *Martens, Bob; Wurzer, Gabriel; Grasl, Thomas; Lorenz, Wolfgang; Schaffranek, Richard. Real Time – Proceedings of the 33rd International Conference on Education and Research in Computer Aided Architectural Design in Europe*, Wien, pp. 415-424

König, R 2010, *Simulation und Visualisierung der Dynamik räumlicher Prozesse*, VS Verlag für Sozialwissenschaften, Wiesbaden

Poirson, E, Petiot, J, Aliouat, E, Boivin, L and Blumenthal, D 2010 'Study of the convergence of Interactive Genetic Algorithm in iterative user's tests: application to car dashboard design', *Proceedings of IDMME - Virtual Concept 2010*, Bordeaux