Evolutionary form design: the application of genetic algorithmic techniques to computer-aided product design

This item was submitted to Loughborough University’s Institutional Repository by the/an author.


Additional Information:

• This is a conference paper.

Metadata Record: https://dspace.lboro.ac.uk/2134/14064

Version: Accepted for publication

Publisher: © Professional Engineering Publishing

Please cite the published version.
This item was submitted to Loughborough’s Institutional Repository (https://dspace.lboro.ac.uk/) by the author and is made available under the following Creative Commons Licence conditions.

You are free:

• to copy, distribute, display, and perform the work

Under the following conditions:

**BY:** Attribution. You must attribute the work in the manner specified by the author or licensor.

**Noncommercial:** You may not use this work for commercial purposes.

**No Derivative Works:** You may not alter, transform, or build upon this work.

• For any reuse or distribution, you must make clear to others the license terms of this work.
• Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the Legal Code (the full license).

Disclaimer □

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/
Evolutionary form design: the application of genetic algorithmic techniques to computer-aided product design

I J GRAHAM, R L WOOD, and K CASE
Department of Manufacturing Engineering, Loughborough University, Leicester, UK

Summary

This paper introduces the current stage of research into the development of a CAD tool that uses evolutionary techniques to assist designers in creating the form of products. A Genetic Algorithm (GA) has been combined with a commercial CAD solid modelling system. This initially enables the creation of a set of apparently random objects. These objects are then subjected to a selective breeding programme, at the hands of the user and also guided by pre-set internal, or environmental, factors. The user gives each object a score, or objective function, influencing which objects are 'fittest', and more likely to become parents of the next generation.

The intention is that, through the co-operation of the user and the pre-set environmental factors, the forms on the screen progressively become more than an abstract collection of geometric primitives. On a primary level, the system can provide the inspiration for aesthetic features and characteristics of products. Further work may develop the potential for a new design methodology. The challenge will be to make the concept genuinely useful, and to do this the outcome of genetic manipulation needs to be predictable, to the extent that desirable features from objects are reproduced in the next generation of objects. The key to this is the way the genetic shape defining data is stored and processed, and is the major focus of this continuing research.

1 Introduction

The collaborative method of user selection is often used in conjunction with genetic techniques [1] for producing computer art. In other cases the evolutionary process is entirely automatic, with existing GA based systems finding mechanical and structural design solutions to meet general user-specified constraints [2]. A novel cell division modelling system [3] has been developed specifically to remove problems associated with using B-Rep and CSG representation for GA manipulation. In common with these examples of related activity, this research looks at taking a step on from traditional design techniques [4].

The inspiration for this research has stemmed from an interest in evolutionary computer programming, Genetic Algorithms and the like [5], and in Computer Aided Design, especially as a concept modelling and development tool for consumer products. At present,
CAD techniques concentrate around the stages of design following conceptual design. The industrial designer uses a sketch pad, a practiced hand and a selection of pencils and markers, perhaps moving on to cardboard, clay, and other physical media, to establish the shape of products. Although a designer may draw inspiration from other objects, natural forms, art etc., it is very much down to the individual designer to create pleasing forms using experience and artistic ability. Existing CAD modelling systems can be useful during this process, allowing a designer to experiment with form quickly and easily. The initial aim of this research is to develop an interactive tool, which will assist designers to a greater extent, during the conceptual stage of aesthetic design.

2 The Evolutionary Form Design (EFD) System

The EFD system is both evolutionary and user driven, with the balance between the two being adjustable. For example, the system can calculate the relative volume of objects and could automatically penalise 'heavy' objects. Some constructional properties are considered automatically, for example, an object that consists of several non-attached parts is given a low objective function, in order to encourage coherent objects. These and other multi-objective factors all combine to create the environment in which the objects evolve. User defined constraints, like object and primitive size limits, number of primitives per object etc., are set using control files, but will eventually be accessed by a complete user interface. Once environmental factors and initial limits have been set, the user has only to provide ratings on each new generation of objects. Although reproductive controls, such as mutation probability and cross-over type (the way in which parent's genes are combined during reproduction), can be altered during an active session. For example, reducing the mutation probability lessens the chance of losing a desirable feature that has evolved over a period of time [6].

The screen shot in Figure 1 shows an example of a first generation of objects. These objects can be viewed individually, rotated etc., before the scores are entered in the box in the lower left corner. Incidentally, it is also possible for the user to participate to a greater degree, by manipulating objects using conventional CAD techniques.

Figure 1 - Screen shot showing first generation objects
The CAD software used is EDS' Unigraphics. Unigraphics provides a 'User Function Development Environment' called UG-Open API consisting of user callable 'C' functions that access the Unigraphics systems. At present, the GA, also written in 'C', runs independently, producing a set of files for each generation. These are read by UG-Open functions which generate and display the objects, before prompting the user for ratings. The objective functions for each object are returned to the GA via the set of 'shared' files. At a later stage, the GA will be fully integrated with the UG-Open functions, allowing a rapid response time between generations.

3 Data Structure

The way in which data is stored is fundamental to the behavior of a genetic algorithm. This, and the types of reproductive techniques selected for crossover and mutation, dictate how genetic data is recombined in subsequent generations and, along with the parent selection method, control the behavior of the system as a whole. The information that defines a geometric primitive and its interaction with surrounding primitives, is made up of six parts. These six parts, which correspond to six chromosomes, each with either one or three segments, are summarised in Table 1.

<table>
<thead>
<tr>
<th>Chromosome</th>
<th>Segments</th>
<th>Numeric range</th>
<th>Decoded to:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>1</td>
<td>0 to 3</td>
<td>block, cylinder, cone or sphere</td>
</tr>
<tr>
<td>Origin</td>
<td>3</td>
<td>0 to 50</td>
<td>local x, y, z co-ordinates</td>
</tr>
<tr>
<td>Sign</td>
<td>1</td>
<td>0 to 3</td>
<td>create, add, subtract or intersect</td>
</tr>
<tr>
<td>Direction</td>
<td>3</td>
<td>-1 to 1</td>
<td>x, y, z vectors, restricted to integers</td>
</tr>
<tr>
<td>Shape</td>
<td>3</td>
<td>0.01 to 1</td>
<td>proportions e.g. length, width, height</td>
</tr>
<tr>
<td>Size</td>
<td>1</td>
<td>1 to 100</td>
<td>multiplier value</td>
</tr>
</tbody>
</table>

Table 1 - Summary of genetic data structure

The 'sign' chromosome controls the interaction with the other primitives that make up an object. The four operators are as follows: A created primitive is displayed, but does not physically join with the other primitives in the object. An added primitive is united with the primitives it touches. A subtracted primitive is removed from the existing primitives. An intersected primitive causes only the material that is common to the current and the existing primitives to be kept. The order in which primitives are introduced is therefore very influential. Incidentally, the first primitive is always 'created', this is achieved by overriding the individual's sign gene.

3.1 Objects formed from 'Teams' of members

Two ways of organising this genetic information have been examined. In one method, each member's genes contain one set of the six chromosomes described above, and are decoded to produce a single primitive. In the language of genetics, the interaction of the genotype with its environment forms the phenotype, in this case, an individual geometric primitive. Several members are then grouped together to form each object.
So far no method has been devised to define the way these ‘teams’ group together; the first five primitives produce the first object, the second five produce the second object etc. So to produce a set of ten objects, 50 members are needed. The fitness function assigned to an object is shared by the individual members. Without any team-forming control, very little visual inheritance could be identified in the offspring and analysing trends and patterns was difficult. To be effective this method will need a set of rules and routines, enabling meaningful collaboration between members. These rules will also be subject to the assigned and internal fitness functions causing the interactions to evolve in parallel with the individuals. These are advanced concepts, even within the field of specific GA research, and so will form the subject of future investigations.

3.2 Single member objects

In order to display inheritance and continuity at this early stage in the research, a second method has been temporarily employed. The storage method has been simplified to make each object the result of one member’s genetic data (each genotype contains five sets of the six chromosomes described above, making 30 chromosomes in all). In this case, the whole object is therefore the phenotype. This method has immediate benefits, making the whole system conceptually simpler, and allowing direct comparisons between parents and offspring. Figure 2 shows two family trees, displaying the inheritance of features, especially in the second picture. Examples of gene dominance can also be seen, along with the variation of objects possible from similar sets of parents. These particular examples show a high degree of variability, which may or may not be desirable. This can be altered to suit the application by changing variables like the mutation probability and cross-over type.

Figure 2 - Two family trees of third generation objects

This type of data structuring works well as long as the user supplied scoring is the primary fitness consideration. However, when more environmental factors are included and the reliance on user ratings is lessened, as is envisaged for future work, the large amount of genetic data per member will create a lot of evolutionary inertia within the population, making evolution slow.
4 Conclusions

These results have shown initial promise, requiring very little experimentation with GA operators. Although the style of early objects are limited to that of the examples shown in Figure 3, interesting shapes have been produced and could be related to a wide range of consumer products. Populations have not yet passed the 3rd generation and the expectation is that objects will appear less 'geometric' as generations progress. If this is not the case, the situation may be improved with larger numbers of primitives per object, but only if these were added gradually throughout the evolution process.

![Figure 3 - Four examples of evolved objects](image)

The GA is not fully integrated with the output software, at the time of writing. An objective measure of the system, which would probably require producing 20-30 generations, has not been carried out yet. When the system is fully integrated it will take just a few seconds to produce each generation. An evaluation could be carried out by deciding on a shape, or product, and then trying to produce the shape in mind.

Further work will also involve creating more internal environmental objectives allowing user defined functional and mechanical design constraints. This would move the work on from an aesthetic tool to a system capable of application to a wider range of design problems.

5 References