Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium 2009, Valencia
Evolution and Trends in Design, Analysis and Construction of Shell and Spatial Structures
28 September – 2 October 2009, Universidad Politecnica de Valencia, Spain
Alberto DOMINGO and Carlos LAZARO (eds.)

A comparison of methods for using genetic algorithms to guide parametric associative design

Peter VON BUELOW

University of Michigan
2000 Bonisteel Blvd, Ann Arbor, MI 48130 USA
pvbuelow@umich.edu

Abstract

The use of parametric software such as Generative Components (Bentley Systems), Grasshopper (Robert NcNeel), Digital Project (Gehry Technologies - Dassault Systemes) and others, has revealed an ever larger variety of architectural form to designers. Now designers are confronted with the problem of navigating this expanded design space in a productive way. Simple trial and error wandering through the parametric variables is a simple solution, but without a systematic procedure a directed exploration of the design space is unlikely to be achieved. In cases where the primary objective is the aesthetic appearance of the form, traditional computational optimization also gives an incomplete view of design possibilities by focusing too narrowly on structural efficiency as defined by predetermined constraints. This paper proposes three different approaches to design space exploration based on the use of genetic algorithms: 1. subjective selection, 2. combined subjective and objective selection, and 3. totally objective selection carried out without user input. The first and second approaches are shown in this paper by examples, and the steps needed to execute the third approach are outlined. Examples include exploration by individuals as well as designers working in groups. A comparison is made between the three approaches and suggestions for achieving better exploration results are given.

Keywords: parametric associative design, genetic algorithm, evolutionary computation

1. Introduction

More detail was provided in an earlier papers by this author regarding the use of genetic algorithms (GAs) in the exploration of architectural form (von Buelow [7],[8],[9]). Likewise, the potential of form generation using parametric software is also well documented (Aish and Woodbury [1]; Coenders [3]). This paper looks at three different approaches to the use of GAs in selecting values for parametric variables to explore a design space.

- Subjective – selection based on form alone
- Combined – selection based on form + performance
- Objective – selection based on performance alone

The first two approaches are shown with more detail in this paper. An overview of the third approach is given here with more details explained in a second paper (von Buelow [10]).

The examples shown in this paper are based on different student projects carried out at the University of Michigan. In both of the approaches that include a subjective input (i.e. approaches 1. and 2.) a group of students actually participated in parallel using a web interface to connect to the GA program. The completely objective approach which is described generates a pallet of solutions without human interaction.

2. Subjective Approach – Form Alone

The subjective approach represents a scenario where no objective criteria are used to determine the fitness function of the GA. This does not mean that performance criteria are not being considered, but rather values for the objective criteria are not being calculated numerically. The evaluation is dependent on the subjective perception and understanding of the person giving the evaluation. So the outcome naturally varies with the ability of the evaluator-user.

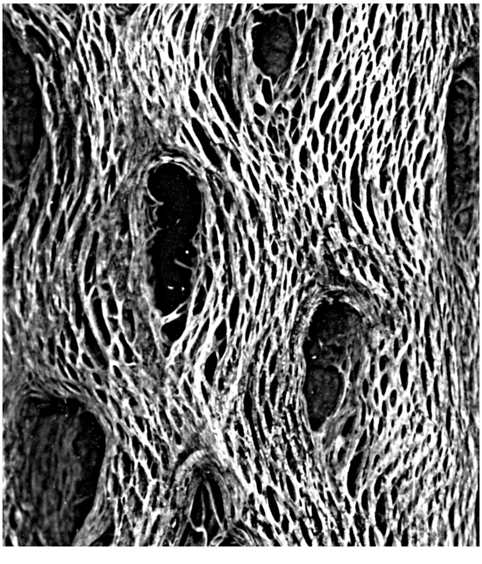

Figure 1: Actual fiber structure of cholla cactus.

2.1. Subjective Example Description

This example comes from the context of a design studio using Digital Project to generate form based on geometric rules. The specific example is take from a student project based on the geometry of the fibrous structure of a dried Cholla cactus (Figure 1). The Digital Project script contained 70 variables (really far too many) which were linked to an Excel spreadsheet. The GA was designed to allow either a group of designers working simultaneously (the class) or the single designer (student) to participate in the design space exploration.

2.2. Subjective Approach Procedure

The window into the design space is afforded by a public web page which displays a population of forms derived from the parametric model. The exploration begins by generating a population of Cholla forms based on random values used for the parametric variables in the model (Figure 2). Given the population of solutions, the students can choose which individuals to breed based on subjective (visual) criteria. By using the web site as the visual interface, multiple designers can use the program simultaneously, thus producing a parallel search procedure. The chosen parents are then bred, producing a child.

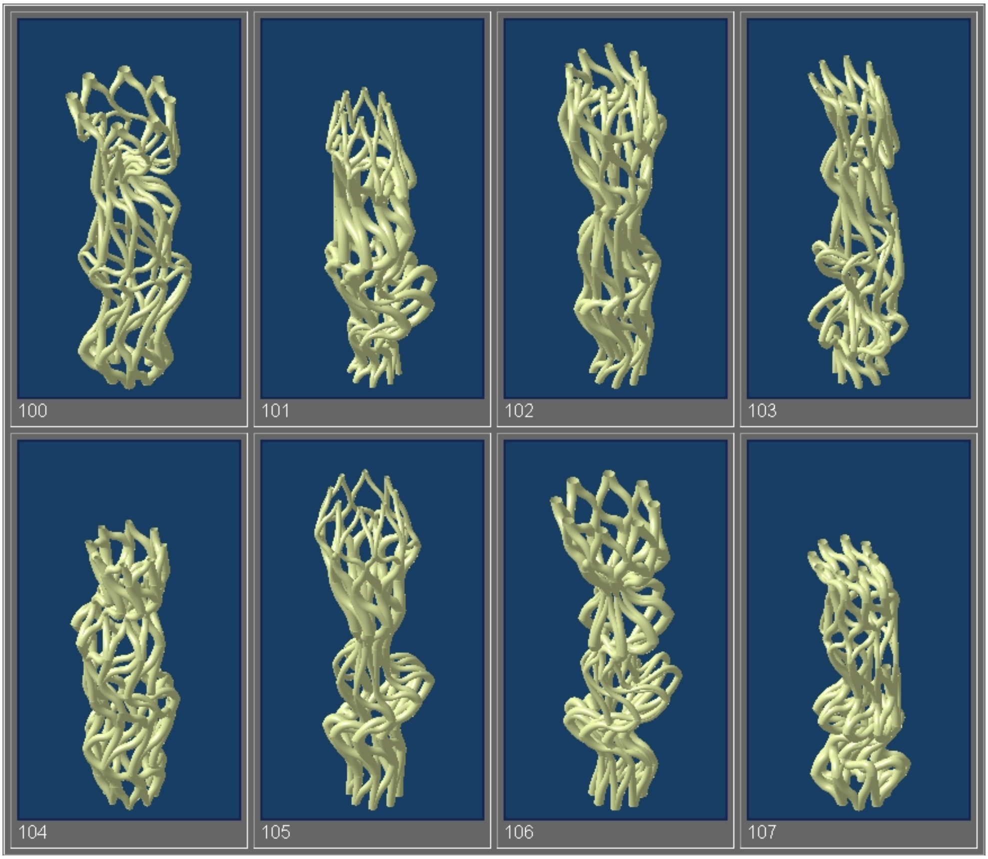

Figure 2: Initial Random Population Used in Subjective Example.

The child is then given form in the parametric software. In turn the child is returned to the population where it competes for breeding and retention along with the rest of the population. Older solutions which remain un-chosen are removed from the population causing a shift towards areas in the design space which correspond to the designers' subjective criteria.

The GA for the subjective example is comprised of typical GA components, but they are not all contained in a single program:

- **Population** – the set of solutions shown to the user on the web page
- **Selection** – made by user with mouse clicks to pick parents
- **Breeding** – using half-uniform-crossover with chromosomes based on csv files
- **Mutation** – the occasional introduction of new random forms to the population
- **Fitness** (ranking) – based on the number of selections recieved and age
- **Generation** – static, (approximately one in, one out)

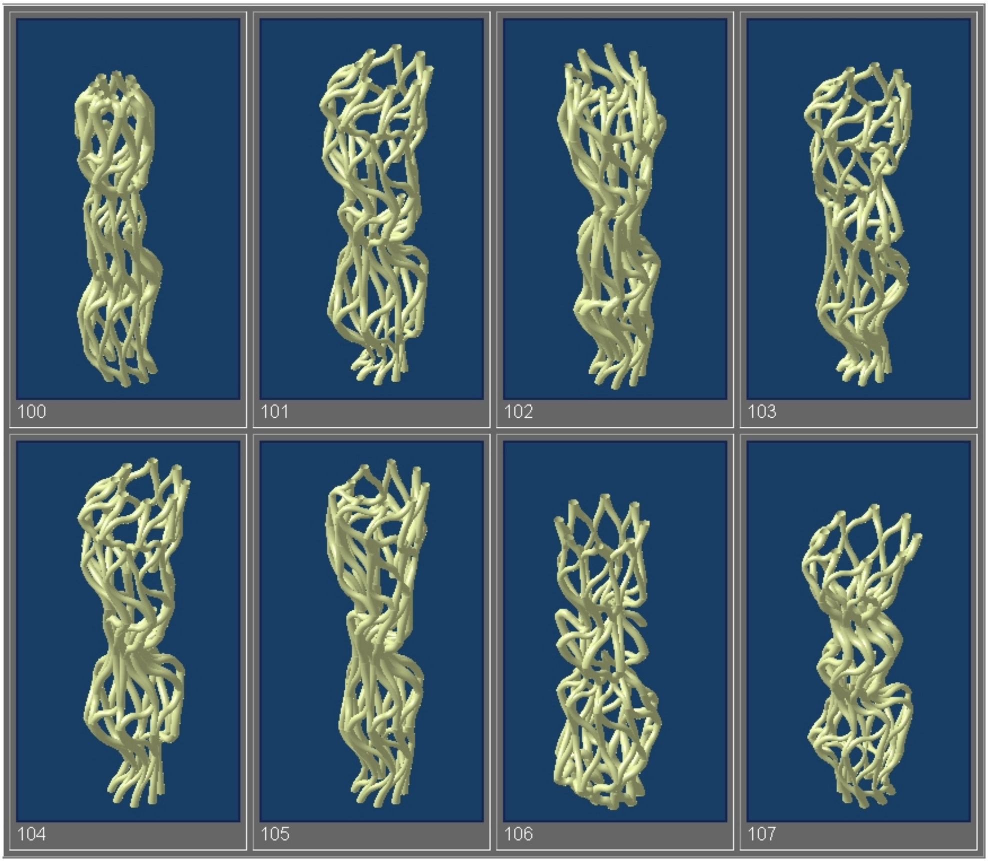

Figure 3. The population using subjective breeding after about one hour.

The population size needs to be kept relatively low (less than 20) so that visual comparisons can easily be made. The common interface for selection is a web page. This allows a large group of users to see the same population simultaniously and make selections interactivly. Image files and csv (Excel) files are stored on the site. Parents are selected for breeding with mouse clicks. The corresponding csv files are then submitted to a program on the web server which uses half-uniform-crossover to produce children (Eshelmann [4]). The program also has user selectable options to mutate a single parent or produce a new random individual.

The number of iterations with this method is of course limited by the human interface. Nonetheless, within about an hour of use certain characteristics did begin to emerge in the population (Figure 3). By using the web interface, a large number of people could potentially be allowed to participate in selecting and breeding individuals simultaneously which could evolve forms more quickly, but with no objective fitness criteria, the contents of the population cannot be 'optimized' but instead, like natural populations, it would always be adapting to current pressures of selection.

3. Combined Approach - Form + Performance

This approach allows for the combination of both objective fitness functions and subjective selection. The objective performance criteria can affect the GA in a few different ways. One way would be to simply make the designer aware of the performance values during selection for breeding. The degree to which the performance influences the selection is then at least in principle controlled by the designer. Determination of the current population could be carried out in the same way as in the subjective approach or an elitist strategy of keeping the best parents based solely on the fitness ranking could be used. In either case the subjective choice of breeding pairs is still made by the designer.

Figure 4: The Foster Bridge, 1876. Span 36 m

3.1. Combined Example Description

The example of this approach also comes from a class of architecture students at Michigan. The assignment was to redesign a local through-truss bridge. Figure 4 shows the existing bridge design dating from 1879. In past years this problem has been given to the class in different ways either in the form of a traditional design problem or as an exercise using structural optimization software. This year the bridge truss was modeled parametrically using Generative Components (GC). The class of 20 students then used the web based GA in combination with both GC and the finite element analysis program, STAAD.Pro to determine the geometry and weight of each design.

3.2. Combined Approach Procedure

The procedure began with the random generation of a population of truss forms. Again the GA chromosome in the form of an Excel csv file was used as input to the parametric modeler, GC. GC generated both a visual image in the form of a jpg file which was uploaded to the collective web site, and a dxf file of the geometry which was used to input the geometry to STAAD.Pro. The load, support and material information for the bridge remained constant and could be attached to the geometry as a stub. This allowed the students to quickly run each analysis by simply attaching the stub to the geometry and running it in STAAD.Pro. The result of the finite element analysis was the weight of the truss. This was used as the file name for the new design files (weightcsv and weightjpg) which were then uploaded to the collective web site.

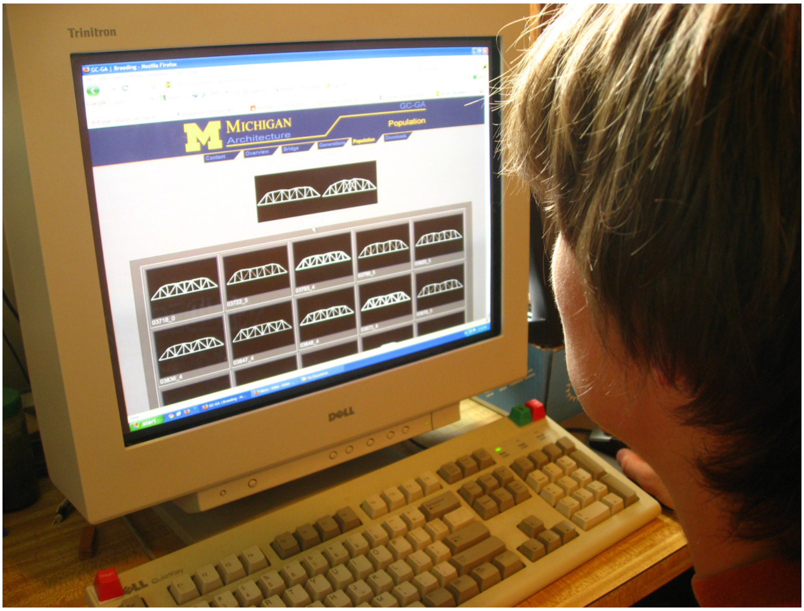

Figure 5: The user web interface for selecting parents and breeding structures. The two trusses at the top of the screen have been selected from the pool and are ready to breed.

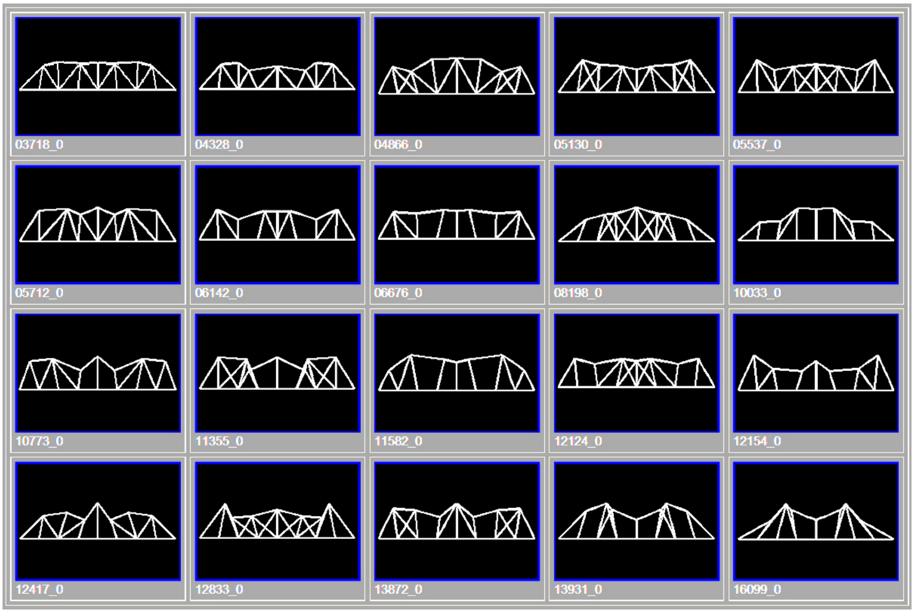

Figure 6: The initial random population. Numbers indicate the weight of each truss.

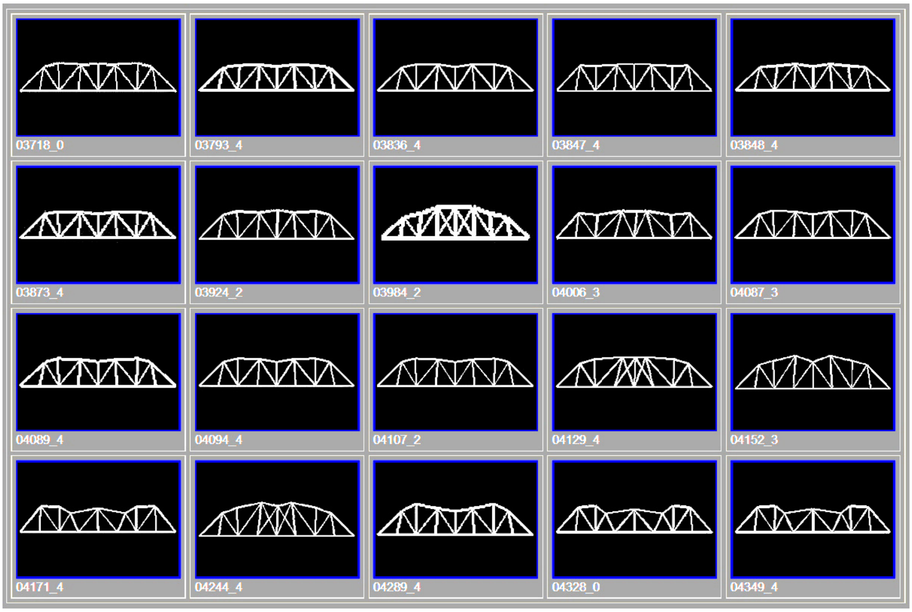

Figure 7: Population 5 - Some degree of convergence, but also variation of solution.

A PHP program on the web server sorted the images by weight (file name) and posted them on the "population" page (Figure 5.). In this trial, actual generations were used (rather than static) and after the full size of parents + children was reached (40) the lightest 20 were retained and the others removed. Figures 6 and 7 show two selected generations in the procedure.

4. Objective Approach – Performance Alone

This approach follows the more traditional GA pattern. No human interaction is possible except, of course, in setting the initial parameters of the parametric model. Examples of this approach to using GAs in parametric design for applications in acoustics, heat gain, structures, etc. can be found in the literature (Coenders [3], Besserud [2], Vanucci [6]). As with the other two methods shown above, this approach would use a conventional Finite Element Analysis (FEA) to determine member forces and required sizes for each parametric variation generated by the GA. An example of this method is given in more detail in a second paper (von Buelow [10]), so only the outline of the method is given here. Also numerous other investigations have been made in the past by the author using a parallel GA and FEA to explore structural form but without the inclusion of parametric associative modeling (von Buelow [7] [8] [9]). In the earlier examples, the process was executed on a Linux cluster using specially coded routines (i.e. no commercial software). The primary difference proposed here is the use of a parametric modeler to develop the geometry. This also means generally the use of commercial MS-Windows based software such as Generative Components and STAAD.Pro. The two methods discussed above can be seen as prototypes of this process since they follow much the same flow chart. However in this last case, the entire process needs to be controlled by a program, whereas in the first two examples, a user executes the commercial software interactivly.

4.1. Objective Approach Procedure

This procedure now under development uses the same components as the other two procedures but combines the programs and steps into one automated cycle. The main advantage in this is that without the need for human interaction many more cycles can be run. But of course the object in the whole process is to make the design space visible to humans and afford an interactive exploration as much as possible. To that end the graphic web interface is preserved. This allows the designer a graphic window into the working process.

The cycle shown in Figure 8 starts with the random generation of solutions. These are initially in the form of csv data files which can be read into MS-Excel and feed into Generative Components (GC) to produce geometry in the form of a dxf file which can be read and analyzed in the FEA software package, STAAD.Pro.

With a population in place the individual parents are randomly paired and bred using a code based on the CHC-GA (Eshelman 1991). This takes place on the server or "master" machine. The result of each breeding is two csv files which are then each sent to a "slave" MS-Windows platform machine. The children csv files can be distributed in parallel to as many machines as are available or until all are sent. Currently, Parallel Virtual Machine (PVM) is used for this data transfer (Geist, et al. [5]).

On the slave machines the csv file is read into MS-Excel so that the variables can be linked to a GC script file. When the GC script is run, the new child geometry is created and both a jpg of the form and a dxf file of the geometry are produced.

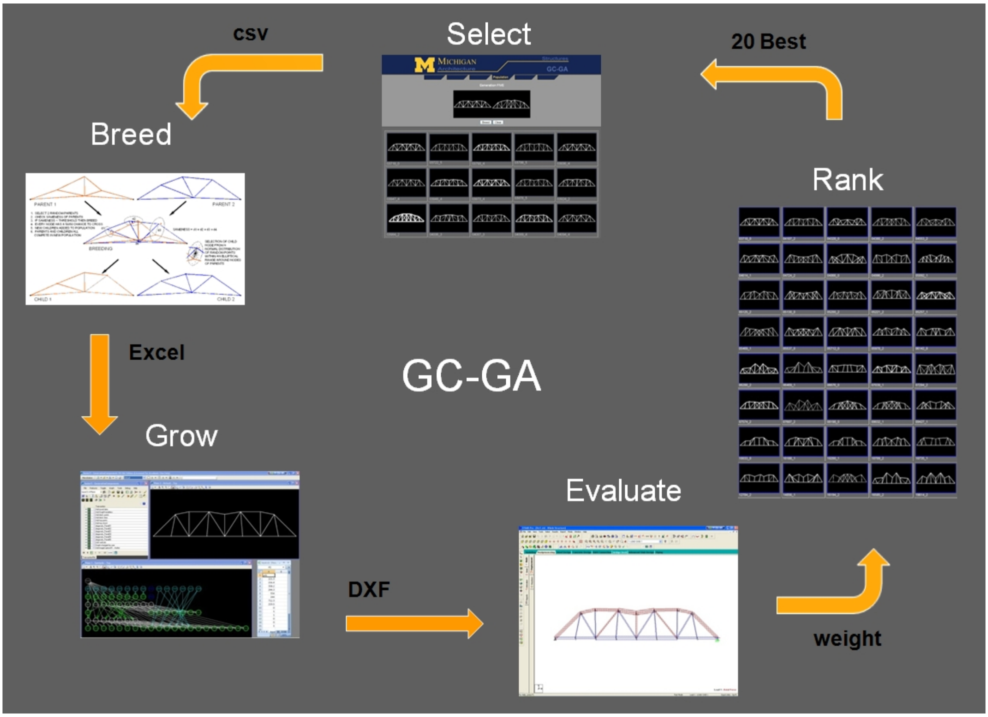

Figure 8. The GC-GA cycle.

The dxf file is then read into STAAD.Pro. OpenSTAAD provides a programming interface to the FEA data file which makes it possible to run STAAD and obtain member design output for the parametric variations of geometry without user input. The member sizes and weight are then added to the csv file, and with the jpg from GC returned to the master machine.

The master machine in this case is a Linux server which hosts the web site and provides the graphic interface. The master program maintains the incoming solutions which are ranked and displayed by population. Because of the way a GA functions, many of the solutions repeat or are very similar. This would ordinarily make looking at all of the solutions difficult due to the large number. To aid the post inspection of the results, the program removes repeated or similar solutions, keeping the ones with the highest performance. In addition a cutoff performance value can be set to remove very poor solutions from the view set. The final filtered set is much more manageable to look through manually and quickly see a range of pretty good solutions. This sifting process has been used with success before by this author in other more traditional GA exploration programs (von Buelow [8]).

5. Comparison of Approaches

The three approaches described in this paper are:

- Subjective – selection based on form alone
- Combined – selection based on form + performance
- Objective – selection based on performance alone

Each offers some advantage under certain circumstances. For a case where an objective fitness function is difficult to identify, the *subjective approach* is really the only option. The subjective example given was unfortunately hampered by the large number of variables in the parametric model. Ordinarily a model of that size might have a tenth the number of variables and this would reduced the complexity to the genetic chromosome, and allow the search to focus more readily. The size of the population and number of generations needed to reach good solutions is typically tied to the length of the chromosome used for the GA. With such a large chromosome the example was really not well suited for interactive use.

The *combined approach* has the advantage that the designer can see both the aesthetic form as well as the pragmatic performance values in making a decision of which direction to explore in choosing parents. The danger here is that the user may be tempted to concentrate on certain attractive solutions too early in the exploration process, and thus miss potentially better solutions. An advantage might be that a team of designers can explore the solutions simultaneously and in an environment where open discussion might lead to new directions or refinement of the problem objectives. In this sense the combined approach might make an effective initial investigation of a problem which could subsequently be followed by the formulation of more complete objective criteria.

With a well defined problem, the *objective approach* offers the best and least time consuming exploration of the design space. By allowing the program to run unattended for a day or two, a much larger range of solutions can be explored. By using the performance limit and filter, a final set of 'pretty good' solutions can be compiled that are of a manageable size to be reviewed by a designer. Ultimately this gives the designer a powerful tool that can be used in the exploration of the wide range of solutions generated with parametric modeling software.

Acknowledgements

The author wishes to thank the Michigan Architecture students who participated in the exploration trials. Also, thanks needs to be expressed to my college TCAUP which provides a 100 node parallel computer cluster for computational research.

References

[1] Aish R. and Woodbury R., Multi-level Interaction in Parametric Design, in: *Smart Graphics*, proceedings of the 5th International Symposium, SG 2005, Frauenwörth Cloister, Germany. 2005.

[2] Besserud K and Cotton J., Architectural Genomics, in *Silicon+Skin > Biological Process and Computation*. Proceedings of the 28th annual conference of ACADIA, Kudless A, Oxman N and Swackhamer M (eds.), 2008, 238-245.

[3] Coenders J.L., Interfacing between parametric associative and structural software. in: *Proceedings of the 4th International Conference on Structural and Construction Engineering*,. Xie M, Patnaikuni I (eds.) Melbourne, Australia, 26-28 September 2007.

[4] Eshelman L., The CHC Adaptive Search Algorithm: How to Have Safe Search When Engaging in Nontraditional Genetic Recombination, in: Foundations of Genetic Algorithms. Morgan Kaufmann Publ., San Mateo, California, 1991.

[5] Geist A. (et al.), PVM: Parallel Virtual Machine. A Users' Guide and Tutorial for Networked Parallel Computing. The MIT Press, Cambridge, Massachusetts, 1994.

[6] Vanucci M. Pluri-Potential Branching System, in *Silicon+Skin > Biological Process and Computation*. Proceedings of the 28th annual conference of ACADIA, Kudless A, Oxman N and Swackhamer M (eds.), 2008, 354-361.

[7] von Buelow P., Breeding Bridges: Genetic based form exploration, in: *IASS 2006 International Symposium: The 8th Asian Pacific Conference on New Shell and Spatial Structures*. Beijing, China. October 16-19, 2006.

[8] von Buelow P., *Genetically Engineered Architecture: design exploration with evolutionary computation*, VMD Verlag Dr. Müller, Saarbrücken, 2007. (also at http://elib.uni-stuttgart.de/opus/volltexte/2007/3160/pdf/PvB_diss.pdf)

[9] von Buelow P., Suitability of Genetic Based Exploration in the Creative Design Process, in: *Digital Creativity*. Routledge, Vol.19, No.1, 2008, 51-61.

[10] von Buelow P. Parametric exploration of discrete structures using evolutionary computation, in: *Evolution and Trends in Design, Analysis and Construction of Shell and Spatial Structures,* Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium 2009, Domingo A and Lazaro C (eds.) Valencia, Spain, 2009.