Genetically Enhanced Parametric Design for Performance Optimization

Peter VON BUELOW
Associate Professor, Dr. -Ing
University of Michigan
Ann Arbor, USA
pvbuelow@umich.edu

Peter von Buelow received a BArch and MS in engineering from the University of Tennessee, and a Dr.-Ing. from the University of Stuttgart. He worked in firms in the US and Germany, and now teaches at the University of Michigan.

Summary
ParaGen is a tool, or more accurately a method, developed over the past several years at the University of Michigan, Hydra Lab. The method uses a genetic algorithm (GA) to search for well performing geometric solutions to architectural engineering problems. The geometric solutions, which are generated with associative parametric design software, are analyzed to determine performance characteristics. The performance is then used to guide a genetic algorithm which searches for good solutions. Solutions are maintained in an online database which can be browsed, filtered and sorted by the designer as a means to both graphically and quantitatively assess the design space. The tool also accepts interactive participation by the designer in the form of 'parent' selection to breed new solutions based on the designer's qualitative preference.

Keywords: genetic algorithms; performance-based design; parametric geometry; exploration; optimization; parallel processing; database.

1. Introduction

With the advent of associative parametric design software, such as Generative Components (Bentley Systems), Grasshopper (Robert McNeel & Associates), Digital Project (Dassault Systemes) and so on, there has been both an increased interest along with the increased capability to develop more complex, free-form geometries for architectural and engineering applications. Although these new means have exposed for the designer a wide range of form vocabulary, the rich array of geometric possibilities is usually uncoupled from performative quantities of the solutions, thus rendering the decision making process more difficult.

ParaGen is intended to aid the designer by both supplying performance data along with the geometric variations, as well as providing a means of searching the design space either through programmed objective functions or by "on-the-fly" user selection. The ParaGen process works by cycling through the following steps:

- Select (either with human interaction or based on a formula)
- Breed (using a genetic algorithm (GA) program)
- Develop (the geometry using parametric software)
- Evaluate (with simulation software, e.g. FEA, Ecotect, etc.)
- Rank (sort the results for exploration and selection through a web interface)

Figure 1 shows a flow chart of one complete cycle. ParaGen will continue to run in this cycle until interrupted by the user or until a set number of new children have been generated without improvement (indicating stagnation).

The ParaGen process makes use of a web server and a cluster of Windows PCs which run the parametric and analysis software. In addition there is an open web interface that can be accessed throughout the design process by any web browser. Together this provides the designer with an effective means to explore large, complex design spaces typical in parametric design.

2. The ParaGen Cycle

As briefly outlined above, the ParaGen method makes use of an array of programs running on both a web server and PC client machines. Typical to genetic based search methods, the entire process is computationally intensive. Also, because the method makes use of commercial software on Windows based PCs, a single cycle can take from several minutes up to over an hour. In order to obtain faster run times, the ParaGen method can easily be run in parallel. Any number of client machines can attach at the same time to the web server and process the individual solutions locally using parametric and analysis software on the local client. The individual solutions are then uploaded to the web server where they are stored in a SQL database along with image and data files. Progress and results can be monitored by individuals or groups of designers simply by connecting to the web site where the solutions are displayed by the server. Because all of the solutions are saved in a relational database, designers can also filter, sort and retrieve the solutions by any of the parametric variables or by analysis data or combinations of either. In this way the method becomes even more useful as an exploration tool of the solution space.

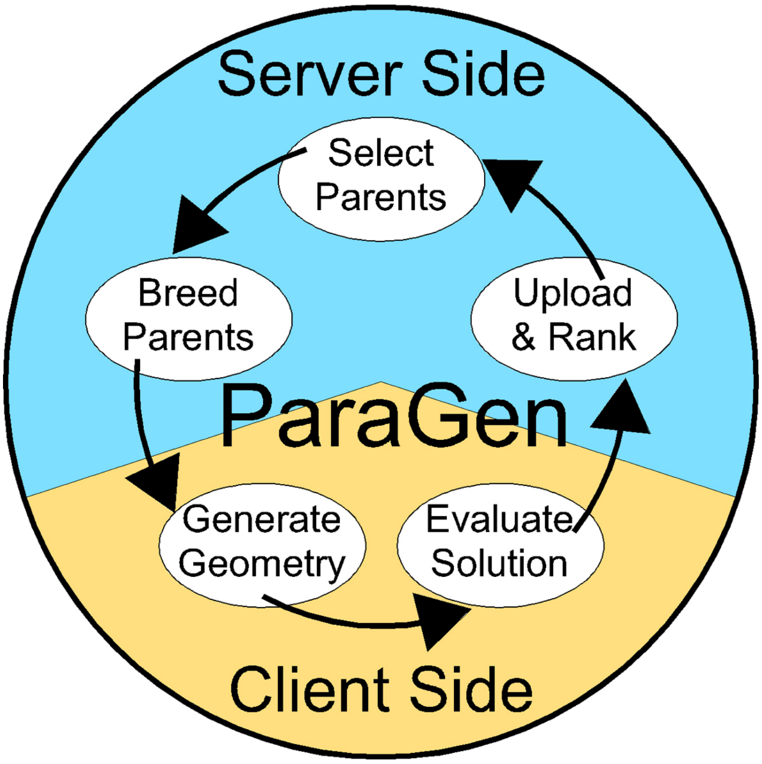

Fig. 1: The Flow Chart for the ParaGen Procedure.

2.1 Selection of Parent Solutions

The process begins with the selection of a pair of parent solutions from the database which are then bred to produce a new child solution. Selection can be made either algorithmically by the program based on performance fitness values, or interactively by the designer using the web interface.

2.1.1 Algorithmic Selection

There are actually a few different selection algorithms depending on the stage of the run.

To jump start the beginning of a run, there needs to be an initial population before selection can begin. This initial population is normally generated randomly. The GA program that runs on the server can produce random sets of input data. The random values are generated within maximum and minimum limits and at appropriate step size for the variables used in the parametric software. Each random data set is sent to a PC client where it runs through the complete cycle, and is returned to the server with corresponding performance values.

Later in the run, after an initial population has been generated, selection is made based on better performance (better performing solutions have a higher chance of selection). This follows selection procedures commonly used in genetic algorithms. Also, preference can be given in the selection of one parent to recent solutions, and in the other parent to best performing solutions. Since the set of recent solutions can include, both designer bred instances and occasional mutant solutions, this promotes potential infusion of new traits into subsequent generations. The population scheme used is similar to the “steady state” approach found in some evolutionary strategies [2]. The actual populations are produced dynamically during the run by queries to the SQL database of all solutions.

2.1.2 Interactive Selection

There is also the possibility for designers to choose parents interactively through the web interface. Since an image of each solution as well as performance data is provided, designers are able to consider any quantitative or qualitative aspects in making a selection. In this way qualities such as aesthetics or the designer’s intuitive senses of the overall design success can be taken into account. To breed two parent solutions, the designer simply selects the images from the web interface and clicks on the “breed” button. Figure 2 shows a selection ready to breed.

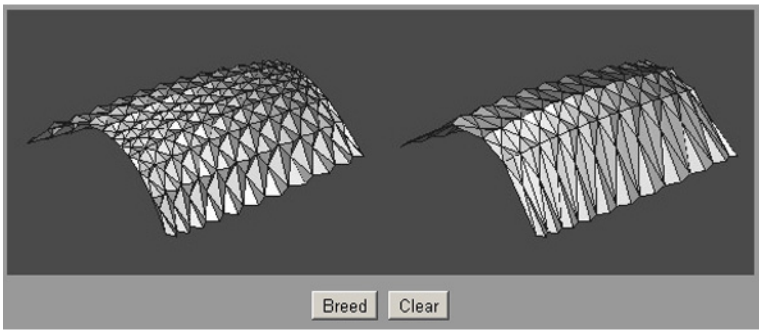

Fig. 2: Interactive breeding of two parent solutions by designer.

2.2 Breeding of New Solutions

The breeding procedure normally functions with two parents, but it can also generate a mutant variation of one parent, or a completely random parent with no prior input. There is also a check made to prevent duplication of existing children. This eliminates repeated processing of the same solution and makes the entire ParaGen process more efficient.

2.2.1 Two Parent Breeding

By whichever way two parents are selected, either algorithmically or interactively, the breeding is the same. Here a variation of half uniform crossover (HUX) employed by Larry Eschelman in his CHC-GA is used [1]. Each variable value is treated as a real number allele on the genetic chromosome. After selecting which values will be crossed, care is taken to preserve the maximum, minimum and step range eventually required for each variable when applied to the parametric modelling software.

2.2.2 One Parent Breeding

If a single parent is selected, the breeding procedure produces a mutation based on that parent. A similar procedure to HUX is followed except the chosen alleles are replaced by randomly generated values rather than crossed with another parent. Mutants can be generated interactively by the designer, but also occasionally occur with algorithmic selection.

2.2.3 No Parent Breeding

In the case where no parents are selected, for example at the start of a run, all of the values are randomly generated. Nonetheless, even in this case, random values are bracketed to fit within the range required for each variable used in the parametric modeller.

2.3 Developing the Geometries

Once a new child data set has been created, it is downloaded from the web server to a client PC which contains the parametric modelling and analysis software. In the examples which follow, Generative Components by Bentley Systems is used to develop the geometry. As noted earlier, ParaGen is conceived to be able to make use of any commercial software or combination of software packages to run the various steps of the process. Since Generative Components accepts Excel files to provide the variable values, the input data set is passed in Excel format.

The script used by the parametric modeller is of course fundamental to the ParaGen exploration since the parameters of the possible geometries that can be reached in the exploration are coded in the parametric modeller script. Therefore, for a thorough exploration of the design space, the script needs to be written to allow a wide range of the different variables.

Once the specific solution has been generated by the parametric modeller, the geometry needs to be exported in a format that can be read into the specific analysis software being used. It is also useful for the later qualitative evaluation by the designer to capture descriptive views of the geometry which can eventually be displayed on the web interface for comparison with other solutions.

2.4 Evaluation of the Solutions

Depending on the complexity of the problem there may be one or several evaluation steps. In this phase the solution is further developed in some simulation software, and quantitative data is collected that will both aid in the decision process as well as direct further exploration through fitness to the objective set in the GA. Often the simulation also offers visual cues to the performance. For example a finite element analysis that determines system stiffness or internal stress levels may also size members and provide a VRML or other 3D image that includes actual member sizes. By saving these qualitative images along with the quantitative values found in the analysis, the designer will later be able to engage the problem at a higher level, and consider both objective and subjective objectives when comparing the range of solutions with the web browser.

After all of the simulations have been run and the evaluation data has been collected, both the qualitative images (or animations or sound files) as well as a file containing all of the quantitative values, are uploaded to the web server.

2.5 Ranking the Solutions

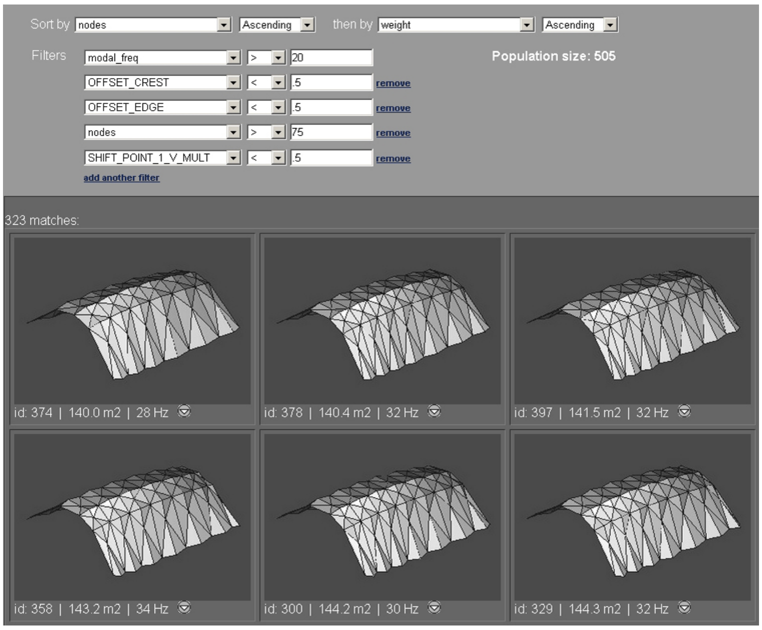

Fig. 3: ParaGen web interface showing the use of multiple sorts and filters to explore solutions.

Once all of the files and data belonging to a solution have been uploaded to the server, the data is placed in a SQL database with an ID tag that links all of the associated qualitative files. The database is accessed through a php program which associates the qualitative and quantitative data through the graphic interface of the web page. Using the web page interface, the designer can both filter and sort all of the collected solutions by any, or any combination, of either the parametric input variables or performance values collected from the simulation software. As shown in figure 3, the web page displays both a small image and key performance values for each solution. By clicking on any solution, a larger image along with all data values is displayed, as shown in figure 4. In this window, links to all other associated files are provided. Since only one instance of each solution is allowed, the widest possible array of different solutions is displayed. Although duplicates are prevented, convergence can still be recognized by the slowed rate of improvement toward the fitness function (objective), and by the visual similarity shown in a sort by best performing values.

Usually, after the procedure has generated a significant number of solutions based on the algorithmic selection, the exploration of the solution set using the filters and sorting functions will suggest possible design directions which the designer can push further with interactive selective breeding. It is also possible to produce Pareto front plots by choosing pairs of independent variables.

With the selection of a new pair of parent solutions for breeding, the procedure starts over with a new cycle.

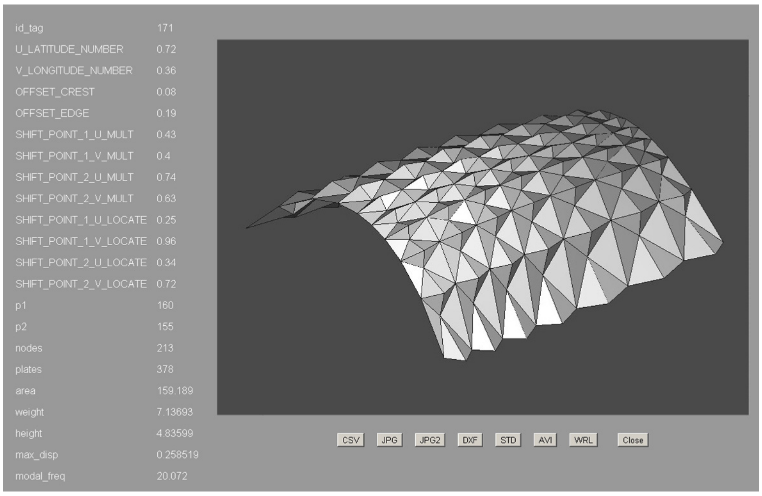

Fig. 4: Detailed view from web interface showing all input variables and performance data, along with access to files which can be retrieved.

3. Example Runs

To date several examples have been run using ParaGen as a design exploration tool. Due to space limitations in this extended abstract, the interested reader is referred to recent papers which cite specific examples [3] [4].

4. Conclusions

This paper has given an overview of the functionality of ParaGen, a conceptual design tool which allows the exploration of parametric geometries based on performance criteria. The procedure which includes a web server and client PCs running various Windows programs, was discussed and options illustrated. The flexibility of ParaGen to run either as a pre-programmed optimization tool or as an interactive exploration tool was detailed. Further development of the system is currently underway with cooperative input of different users.

4.1 Acknowledgements

The author would like to thank Taubman College for support of the ParaGen project. Also the continued interest and coding support with Generative Components by Michela Turin at TU Delft has been key in moving the project forward.

5. References

[1] 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. pp. 265-283

[2] SARMA J., DE JONG K, “Generation Gap Methods” in: *Handbook of Evolutionary Computation*, Oxford University Press, 1997, pp. C2.7:1-5.

[3] TURRIN M., VON BUELOW, P., STOUFFS, R., KILIAN A., “Performance-oriented design of large passive solar roofs. A method for the integration of parametric modeling and genetic algorithms”, in: *Proceedings of eCAADe 2010, 28th Conference: Future Cities*. Zurich, Switzerland, 15-18 September 2010, pp. 321-330.

[4] VON BUELOW P., FALK A., TURRIN M., “Optimization of structural form using a genetic algorithm to search associative parametric geometry” in: *Structures & Architecture,* ed. Cruz, P. Proceedings of the First International Conference on Structures and Architecture (ICSA 2010), 21-23 July 2010. Guimaraes, Portugal.