Development of an Intelligent Genetic Design Tool (IGDT) for Use in Architectural Engineering

Peter von Buelow
University of Stuttgart
Institute for Lightweight Structures (IL)
Stuttgart, Germany
pvbuelow@il.bauingenieure.uni-stuttgart.de
pvbuelow@utkux.utk.edu

Abstract

This paper describes the application of an Intelligent Genetic Design Tool (IGDT) in the design of architectural, structural elements. As a computer design aid an IGDT is innovative in its intelligent interaction with the designer. By always submitting multiple solutions for review by the designer, it is less likely to cause design fixation than other optimization techniques, and allows the user greater range in exploring hard-to-code design criteria such as aesthetics. As an example, the design of a cantilever truss is briefly explored. Using the coded optimization criterion of weight, and the designer's non-coded criteria of visual aesthetics and performance, a series of possible designs are explored. The ability of an IGDT to intelligently respond to the designer's preferences in a way that promotes creative thinking on the part of the designer is demonstrated. A final truss design is selected based on the use of the tool. It is concluded that an IGDT offers a significantly different approach to computer aided structural design which has the potential to enhance the user's own creativity in determining a good solution.

1 Creative Design Tool Criteria

In engineering fields, accomplishing an objective with a minimum of effort, either in terms of material, time or other expense, is a basic activity. For this reason it is easy to understand the interest designers have in different optimization techniques. Mathematical, as well as, model based tools have traditionally been employed for such optimization. In recent times, mathematical methods executed on computers have become predominant. Unfortunately, computer derived solutions often obscure the range of possible solutions from the designer by only exhibiting a final, "best" solution. This not only deprives the designer of seeing alternatives, but also tends to promote designer fixation on the single, machine proffered solution (Purcell & Gero, 1996). Naturally, optimization methods can only respond to the objective parameters which are coded into the problem, and as a result, non-coded parameters, such as aesthetics, historic context, or meaning are left out of the optimization process, and ultimately left out of the final design solution. Lost too in most computer optimization, is the variety of information and creative stimulation that comes from manipulating physical models. Genetic Algorithms (GAs) are able to overcome some of the short comings which hinder traditional computer based optimization techniques in application as design tools. They can work to avoid designer fixation by presenting the designer with populations or successive generations of solutions rather than a single end result. By including interactive user selection, GAs can also incorporate non-coded design criteria into the search process.

The idea of generating design forms or shapes using Genetic Algorithms (GAs) is not new. Many people are familiar with Richard Dawkins Biomorphs, described in his book *The Blind Watchmaker* (1986) or the graphic art of Karl Simms and William Latham (Kelly, 1995). In the area of engineering, form is closely coupled with performance. John Frazer is one who has used GAs in an engineering design sense. In an application dealing with sailing yacht design, the GA optimization considered both objective engineering parameters (stability, center of buoyancy, wetted surface area, prismatic coefficient, blocking), as well as designer criteria such as aesthetic appearance, historic tradition, allusion of form, and so on (Frazer, 1995). The Intelligent Genetic Design Tool (IGDT) described in this paper, develops many of these same ideas.

1.1 Analysis Tools vs. Design Tools

In recent years a variety of computer based tools have been developed for the field of architectural engineering and design. Although originally applied to areas of computationally intensive *analysis*, with the ever increasing size and speed of processors, attempts have been made to develop *design* oriented applications as well. In the field of architecture engineering, the requirements of design tools are distinctly different from those of analysis tools. The analysis process can usually be executed in a regular, predetermined sequence of steps. The sequence may be iterative or have various paths based on the particulars of the problem, but there remains one "correct" solution for one stated problem. For example, for a given structural member, with a given load, a particular analysis will

yield one solution for the stresses at some point. The analysis is composed of a sequence of prescribed steps which lead to the solution. Since the analysis is meant to reflect some single existing condition, the solution to the degree that it is correct reflects this same single condition. To the degree that different individuals or different methods show a discrepancy in their solutions, it is assumed that some error is present. In fact, an analysis is usually verified by showing consistency with the solution obtained in another way.

In contrast, the design process is not expected to consistently yield the same solution. Although a designer may follow a sequence of steps, the steps are not self contained, but influenced by factors outside the process itself (the unique background of the designer, stimuli of the environment, etc.). For a given design problem with a given set of parameters, it is certainly *not* expected that any two designers will come to the same solution. One need only look at the results of any design competition to see the variety of solutions that can be proposed. If a competition for a bridge or building resulted in every entry being identical, the effort would be considered a failure. Design implies creative thinking, and creative thinking does not fit a prescribed set of serial steps. Unlike the analysis of a single given condition, a design solution partially defines the condition, and therefore, seldom results in only one possible solution.

An IGDT goes beyond a set procedure of analysis to aid the designer in exploring form finding problems in a creative way. Unlike analysis tools it is not intended to yield one correct solution, but rather to supply the designer with stimulating, plausible directions to consider. An IGDT is intended to be used in the early, form finding phases of a design problem. As such, it deliberately avoids leading the designer to a single "best" solution, but instead follows the designer's lead in exploring the design space.

1.2 Advantages of Evolutionary Systems in Design

In assisting with conceptual design, Evolutionary Systems, and in particular Genetic Algorithms (GAs), contain certain critical characteristics which are lacking in other numerical methods.

- Use of Populations
- Recombination and Mutation
- Wide Search of Design Space
- No Knowledge of the Fitness Function
- Imitation of Human Design Process
- Can Learn from Designer

1.2.1 Use of Populations

Fundamental to a GA is the use of populations of solutions rather than a single best solution. This is compatible with the way ideas are often generated and regarded in early design phases. Multiple, simultaneous ideas are necessary for the dynamic movement of thought associated with creative design. Ideas play against each other leading to further new ideas. As was stated earlier, creative design is typically not a single-path, linear process, but a more complex, multi-path exploration of many possible ideas.

1.2.2 Recombination and Mutation

A GA uses recombination and mutation to generate new solutions to a problem. This is very similar to design tools developed by William Gordon (Gordon, 1961) which attempt to combine diverse aspects of different solutions to achieve a new perspective. In a GA groups of possible solutions are recombined in a like manner. One of Gordan's techniques is to fragment and recombine words and phrases. In Synectics one of the "operational mechanisms" suggested to promote creative thinking is to play with words and phrases and their meanings as a way of making the familiar strange. Similarly, Albert Einstein comments that, "combinatory play seems to be the essential feature in productive thought." (Reisner, 1931). This "play" is similar to the mechanisms of random mutation and recombination used in genetic techniques.

1.2.3 Wide Search of Design Space

Through the genetic operators of mutation and random recombination, GAs are very effective at exploring the complete design space of a problem. Complex problems may contain many "optimum" solutions. Rather than focusing on just one such optimum, GAs will continue to search for other optima as well. In reference to creative design, George Prince stresses the importance of free speculation as a means of enhancing creative thought.

> We believe that as the expert accumulates the specific knowledge that makes him so valuable he also incorporates the accepted certainties that are not really certain. This explains why, historically, so many innovative breakthroughs have come from outsiders rather than from those who are thought to be most expert in the particular field. (Prince, 1970)

Using mechanisms of random mutation, GAs are able to retain a degree of speculation. In this way they can help even Prince's "expert" to consider new possibilities in the design space.

1.2.4 No knowledge of the objective function

The fact that the GA itself operates independently of the objective function, allows it to be controlled, without

programming, by the designer. The GA is steered by means of a fitness function, but the source of that fitness is not necessarily defined by coded objectives alone. The GA can be guided by any means of evaluation, either coded, non-coded or a combination of both. This is very important when considering some hard-to-code, qualitative or subjective design criteria such as aesthetics. Engineering design combines subjective and objective design criteria. The IGDT works on both levels at once, using coded objective criteria in searching for good solutions, and non-coded subjective criteria supplied by the designer's interactive selections.

1.2.5 Imitation of human design process

A GA learns in a way that is analogous to human learning modes. It seeks a solution by considering many options. Like the near random flow of ideas that a designer will sift through at the start of each project in search of usable concepts, a GA processes thousands of solutions, comparing, crossing, recombining, altering, sorting, keeping the best and always scanning the design space for better ideas. The process is not always direct but it is always goal oriented. It is also thinkable that the criteria may evolve as the project matures. As the orienting fitness function changes, the GA automatically adapts and searches in the shifted direction.

1.2.6 Can learn from designer

Like the adaptation of biological systems to their environment, the GA can learn from the fitness information it is fed regardless of the source. Thus, the GA can be guided with criteria that even the designer might find difficult to express. In so far as the designer is consistent in ranking the fitness of solutions, the GA will learn from the designer. By considering the solutions proposed by the GA, the designer too may learn and form a new viewpoint of proposed design solutions. The GA is also able to adapt to changes in the designer's understanding without direct reprogramming from the designer.

2 Characteristics of an IGDT

2.1 Definition of the IGDT Concept

An Intelligent Genetic Design Tool (IGDT) is a tool that is able to dynamically adapt to evolving design criteria, through interaction with the designer, to aid in the exploration of a range of good solutions. It is intended to assist the designer in the early, conceptual design phases. The concept of an IGDT is significantly different from traditional analysis and design programs in three ways.

- Does not require pre-programmed, quantifiable objectives
- Provides intelligent interface - responsive to the user's ideas
- Aids and stimulates the creative exploitation of the design space

Although an IGDT uses optimization, it is distinguished from traditional optimization methods in its ability to adapt to non-programmed fitness criteria learned from user interaction. An IGDT is able to learn by employing genetic operators of recombination, mutation and selection to a population of solutions which evolve based on the selective pressure which can be supplied through human interaction. Because the objectives of an IGDT are not explicitly pre-programmed, an IGDT can be implemented in earlier design phases than is possible with more common analysis tools, without the danger of causing design fixation or prematurely restricting the design process. Many initial design considerations which are qualitative parameters such as aesthetics or complexity of form, have traditionally been defined through examples. An IGDT, which communicates with the user through examples, provides a more natural and effective tool for design than programs that require pre-defined, quantifiable objectives. Also, because an IGDT learns from interaction with the user, it is more easily approached by non-computer oriented users. The concept of an IGDT is applicable to many fields involving design.

2.2 Relation of IGDT to Design Process

Genetic Algorithms (GAs) are used as the basis for an IGDT because they are conceptually very close in operation to the way many designers work. GAs generate populations of individuals in the way that designers create a pool of ideas to draw from. GAs operate on this population by recombining parts of different individuals or altering existing individuals though mutation (Holland, 1975). Designers, too, combine good aspects of various ideas, and alter old ideas to fit new situations. GAs search a wide design space for various good solutions before narrowing the scope to a single best solution. Designers behave similarly by considering several options. Also, because a GA need not have programmed knowledge of the solution objective, the fitness of the individual solutions can be determined by the designer using qualitative criteria. By not requiring the design criteria to be numerically expressed or directly programmed, an IGDT remains flexible, and responsive to changes dictated by the designer.

For design applications involving form, the ability to view and manipulate proposed solutions is essential to the decision making process. In this paper an IGDT is demonstrated using a trussed structural system as an example. The concept presented, however, is not limited to trusses, or even structural systems. The basic concept of an IGDT can be adapted to a wide range of form determining design problems.

3 Mechanics of an IGDT

3.1 Layers of Operation

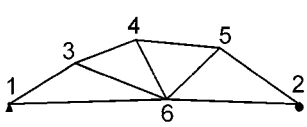
The IGDT functions in two different layers. In the outer layer, The IGDT interacts directly with the user. It proffers solutions to the designer for critique and ranking. The designer's ranking provides the fitness value for the individual solutions.

A second, inner layer operates within the outer layer. The inner layer searches the design space for good solutions to feed to the outer layer. While the outer layer searches for acceptable topologies with the designer, the inner layer finds optimal geometries for each topology. In this way the proffered solutions are reasonably good solutions for the selected topologies.

3.1.1 The Topology GA

The outer layer uses the structure's incidence matrix as a chromosome. The incidence matrix describes the connectivity of the members and joints by recording the start and end joint number of each member. Because the incidence matrix is binary, it lends itself easily to conversion into a binary string chromosome. Figure 1. shows a truss with its incidence matrix and derived binary chromosome.

Because the topology is so easily described as a binary string, a conventional binary string GA is used, patterned after Holland and Goldberg (Holland, 1975; Goldberg, 1989). With the intent of providing less disruption and of preserving longer schemata, single-point crossover is used. At this point no testing has been performed to bear this out.



\	1	2	3	4	5	6
1	\	0	1	0	0	1
2		\	0	0	1	1
3			\	1	0	1
4				\	1	1
5					\	1
6						\

010010011101111

Figure 1: An example truss with numbered nodes, incidence matrix, and defining binary string.

Not every topology produced randomly or by crossover represents a stable, usable truss. For example joints may be left with no connecting members, or members may be left isolated with no connection to the rest of the structure. These cases are simply regarded as still births, and the process that produced them is repeated until a stable solution in found.

3.1.2 The Geometry GA

The geometry GA works with a given topology to find good geometry solutions. The geometry is defined by the Cartesian coordinates (real numbers) of each joint in the structure. The programming mechanics used for the inner layer, are based on the CHC method developed by Eshelman and Schaffer at Philips Laboratories (Eshelman, 1991). The CHC varies from the more traditional GA methods in that it uses a strongly elitist, steady-state selection strategy, and segregates the operations of mutation and crossover. CHC has the advantage of being able to work with relatively small populations of 50 individuals per generation, and still succeed over a wide range of problems. The breeding algorithm used is similar to that employed in a (μ+λ) Evolutionary Strategy (Bäck, Hoffmeister & Schwefel, 1992). CHC is also unique in that it segregates mutation and crossover by running in cycles using crossover only, and then restarting a cycle by mutating the best individual to fill the restart population. Figure 2 shows the overall CHC cycling and Figure 3 shows the GA portion of one cycle.

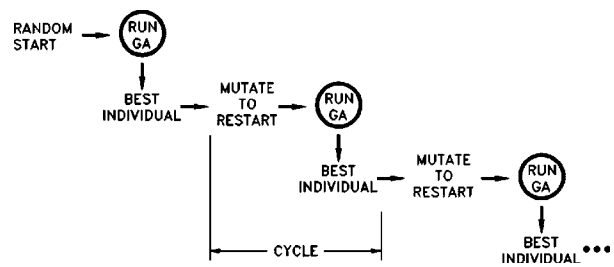


Figure 2: Overall plan of the CHC - GA.

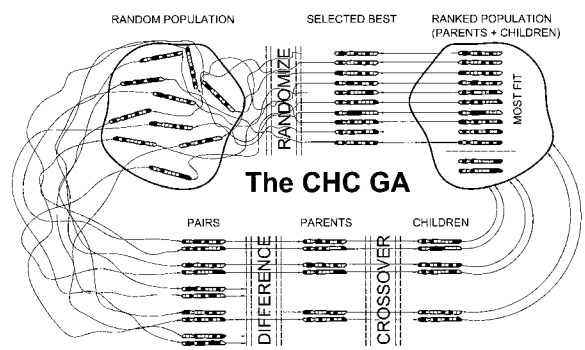


Figure 3: The "GA" portion of the CHC.

Because the joint coordinates are real numbers with a meaningful spatial location, it was decided to incorporate these qualities into the breeding mechanism. During breeding, the joint coordinates of a child is selected from points in a natural distribution about the parent points. Figure 4 shows an example of the breeding process for two trusses.

4 IGDT Operation

4.1 Outline of an IGDT Session

The operation of the IGDT is iterative and can continue as long as the designer finds it productive. Like any dialog, it will reach a point of stability where so little new ground is being covered that continuing is not effective. Because the IGDT interacts intelligently with the designer, repeated sessions at later times can offer different results since the designer's own understanding of the criteria will evolve over time. Therefore, more complex problems may benefit from multiple sessions spaced a few days apart. Within one session the activity can be outlined as follows:

- Problem Definition
- Initial IGDT Proposals
- Designer Selection / Interaction
- IGDT Response
- Iteration of Previous Two Steps...

4.1.1 Problem Definition

The user initially sets constant design parameters which are stored in a data file read by the IGDT at the start of a session. Constant criteria include material constants and properties; topology constants such as symmetry; geometry constants such as support positions or required load points; required load cases and load combinations; analysis method and output specifications. For the cantilever truss example the following problem definition was used:

System	Plane truss
Material	Steel (ASTM A-36)
Sections	Pipe (based on Schedule 40 with continuous size range
Supports	one with fixed x and y coordinates (upper support) one with fixed x coordinate only (the lower support)
Loads	1 Point load of 10 000 LBS at 30 FEET from the supports
Analysis	AISC ASD steel code
Fitness	least weight)

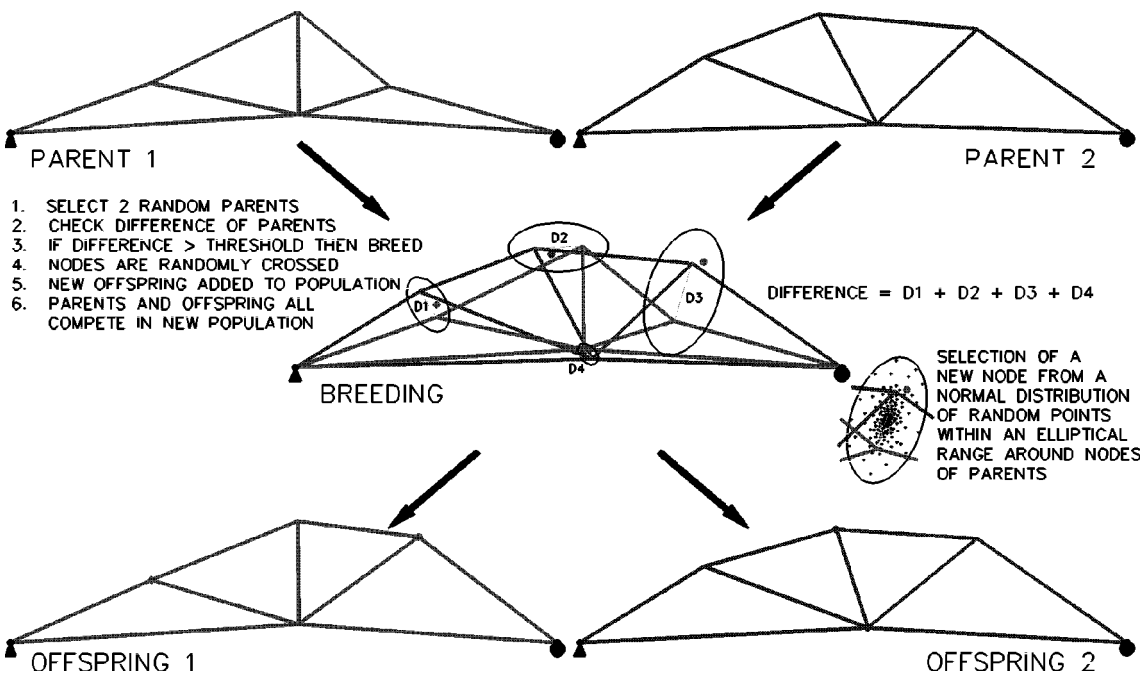


Figure 4: The breeding of two parents and the calculation of a difference threshold.

The program is modularly designed to allow for the easy incorporation of a variety of optional subroutines. The optional subroutines are also chosen at the start of the trial. In the current truss design program, optional subroutines include material (steel or wood), analysis methods (wood or steel codes, Euler buckling, or simple axial stress), loading types and combinations (point loads or self weight). Any joint coordinates can be set to appear positionally "frozen" to the GA. They can be given a pre-specified, set position in the x or y direction or both directions. This is particularly useful in defining supports or specifically located point loads.

Fitness in this example was limited for simplicity to weight, but it is possible to include other criteria as well, such as total number of joints, complexity of joints (number of members meeting at a joint), oversize members, waste from standard lengths, etc.

4.1.2 Initial IGDT Generation

Figure 5. shows the first generation of solutions from the IGDT. These represent either different topologies and/or significantly different geometries. The dashed, members indicate tension, and continuous, members indicate compression. In this example a population of 18 was used. This is seen as an overseeable number for the designer. Each solution is shown in a box and they are all sorted from top to bottom by fitness. The solutions are numbered consecutively from the first generation onward, and can be retrieved from storage at any time. The proposals reflect the inner layer optimization based on the fitness criteria set by the designer, but are intended to display a variety of different good solutions rather than just the best solutions.

4.1.3 Initial Generation with Designer Selection / Interaction

Presented with a palate of proposals from the IGDT, the designer begins the iterative dialog which allows the IGDT to learn the preferences of the individual designer on the specific project being investigated. The designer proceeds by selecting and marking best and worst proposals.

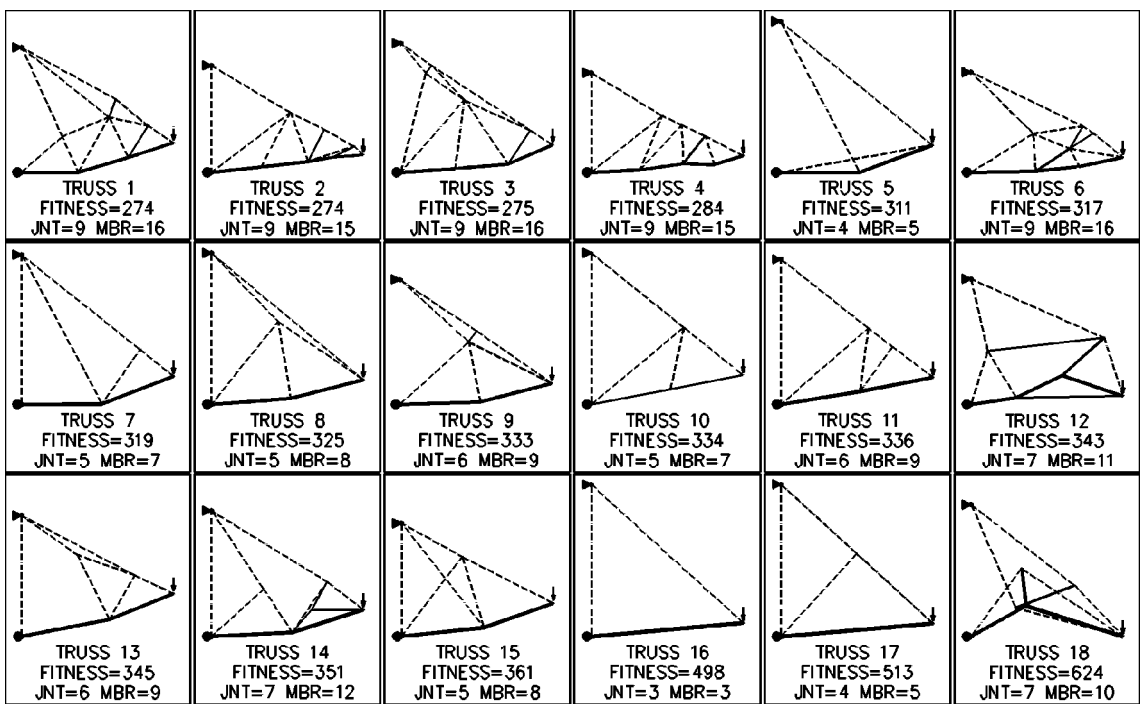


Figure 5: An initial IGDT generation with proposed solutions arranged by the specified fitness of least weight.

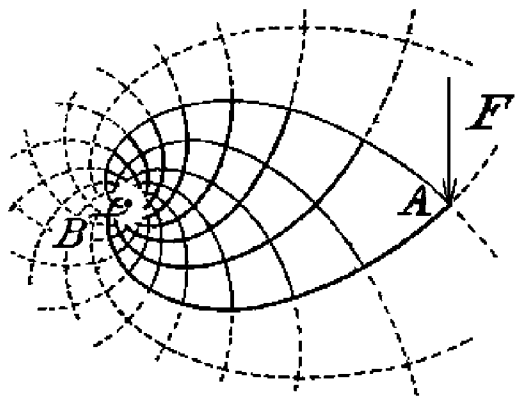


Figure 6: Mitchell's diagram for a cantilever case. (Mitchell, 1904).

Figure 6. shows a Mitchell diagram for an "optimum" solution of a cantilever. Mitchell's support conditions a somewhat different, and he did not consider stability (no buckling). Nonetheless, Truss 1. is very similar in form to this idealized solution. But, for the sake of the example, it will be taken that the designer sees visual and practical problems with this form. From the front view (looking from the loaded point, back toward the supports) the bottom chords of the structure are exposed which might cause weathering problems. Further it will be supposed that the designer desires the visual effect of a clean line in elevation view at the point load with preferably an upwardly curving underside. The designer indicates these preferences by selecting Trusses 2, 4, 12, and 18. The solutions 10,11,16 and 17 the designer considers visually too simple, and marks for deletion. The effect of this selection is to remove the undesirable trusses form the breeding pool, and to increase the likelihood of the selected trusses to breed more often.

4.1.4 Second Generation IGDT Response

Using the designer's input, the IGDT searches for a new set of proposals. First new topologies are found using binary GA operations of mutation and recombination performed on the incidence chromosome shown in Figure 1. The children topologies are then established with good geometries using the CHC-GA. Again the proposals represent either different topologies and/or significantly different geometries, and are sorted and numbered by fitness. Figure 7. shows the IGDT's response to the designer. Unlike the initial proposals of the IGDT, all subsequent generations of proposals are derived from, or influenced by, the interaction with the designer. Some of the same topologies show up, but as noted above, parents are not included in subsequent generations, unless deliberately retrieved by the designer and reintroduced to the breeding population. Upon examining Figure 7. one sees that the form tends naturally toward the form of the Mitchell diagram, but several geometries are now proposed with concave bottom chords.

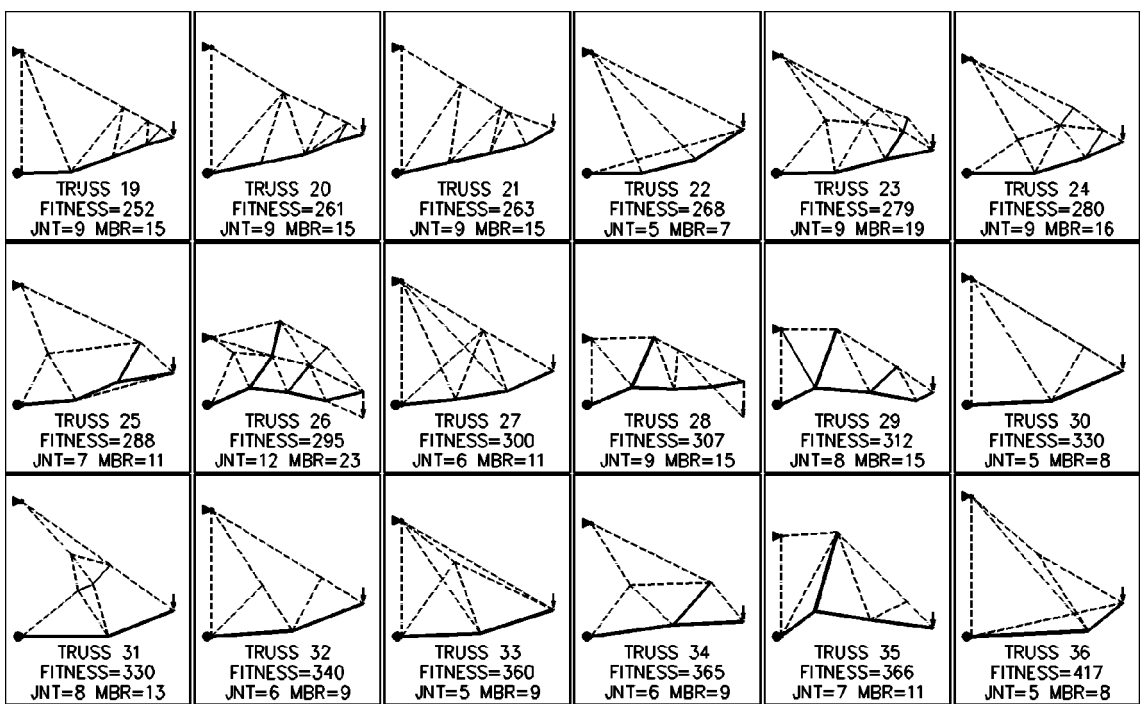


Figure 7: The second generation produced by the IGDT.

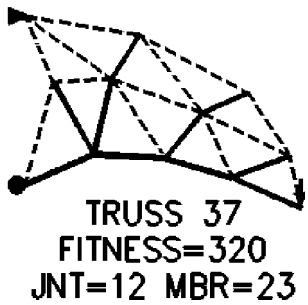


Figure 8: Designer modified Truss 26, introduced as Truss 37.

4.1.5 Third Generation

Figure 9. shows the IGDT's response to the selections made in the previous generation. There are now several new topologies which exhibit geometries with concave bottom chords. The process continues as with the previous generation. The designer indicates positive and negative preferences, and the IGDT breeds the surviving solutions to find new possible solutions that lie in the direction of the designers preferences.

Even at this early stage some of the IGDT proposals can be helpful to the designer. For example, Truss 49 from the third generation (enlarged in Figure 10.) is in the direction being pursued by the designer. It is similar to the designer's own proposal (Figure 8. Truss 37) but offers some advantages of simplicity and economy of material. The most weight efficient solution found by the IGDT (Figure 9. Truss 38) resembles a ridged bottom plate hung in place with tension members. This may indicate a potentially new concept for the designer to consider as opposed to a completely ridged truss.

Thus, the designer can continue to explore possible design concepts with the IGDT. Like any design process, it ends when the designer is satisfied with the solution (or constraints such as time force a decision). In the course of the exchange, the IGDT learns the designer's preferences, on whatever those preferences may be based. The designer also learns as the range of possible solutions is explored. As new sets of proposals are presented, the images will suggest new considerations and concepts to the designer. In the course of a session, it is expected that the thinking and considerations made by the designer in responding to the IGDT will undergo a development of their own. This is precisely why Genetic Algorithms

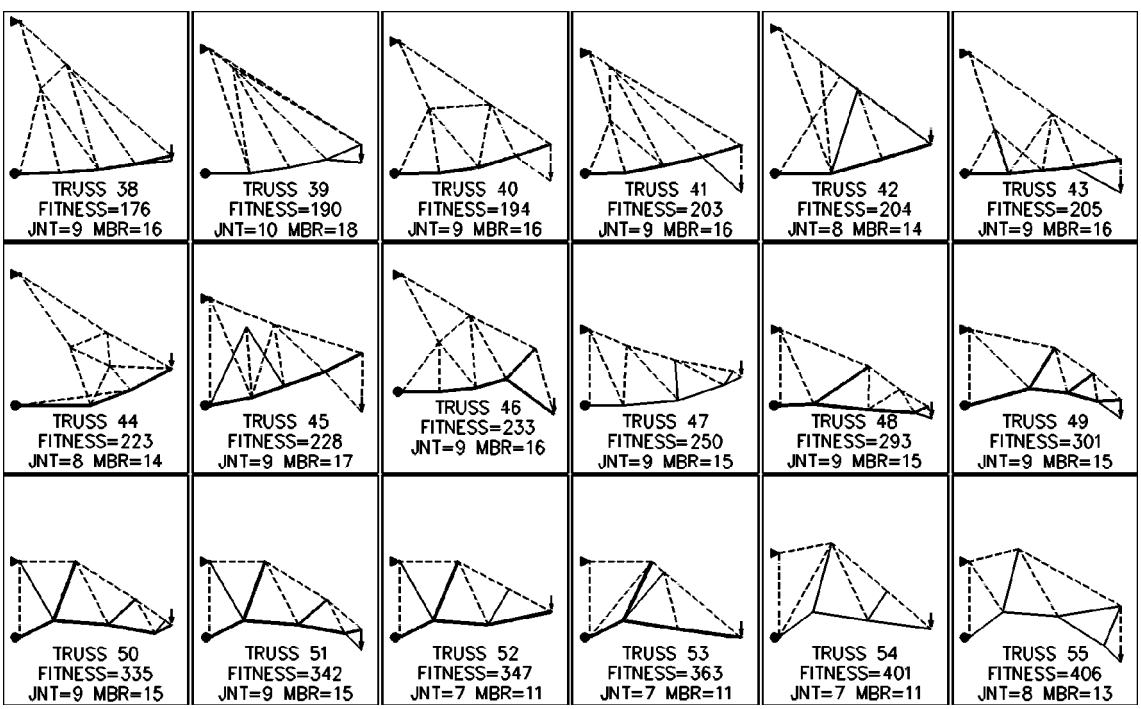


Figure 9: The third generation produced by the IGDT.

work so well in the IGDT. Since GAs do not require any knowledge as to how the selection decisions are made, they are free to follow whatever direction is indicated by the designer. As a result the path toward the solution will not be direct. Old solutions may from time to time resurface under changing criteria. This is not important. What is important is that the design space be thoroughly explored and made apparent to the designer. The success of the IGDT is based more on a thorough search of possibilities than on finding precisely the single, ultimate best solution.

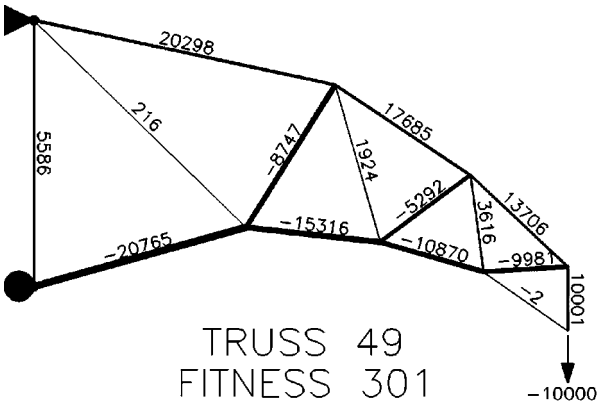


Figure 9: The final solution selected by the designer.

5 Conclusions

Although it would be easy to suppose that an IGDT helps the designer to find a solution which meets the aesthetic and performance criteria while still providing good economy with low weight, this paper does not really try to prove the point. The tool is intended to help the designer in the exploration of structures that are otherwise difficult to assess. A simple, flat cantilever truss is not the best example of this, but is an example that lies within the current limits of the programs development. In addition, the "designer" in this case was the same as the programs author, and the author of this paper - not altogether an impartial test. The problem too, was chosen as one that worked out to some degree. In most cases there is much more wandering in the procedure. But the wandering need not always be regarded as negative, provided that it does not simply circle the same position.

The form chosen by the designer differed significantly from ideal Mitchell diagram form which direct optimization programs would suggest. In the process of developing the final solution, many possible directions were explored, and the designer received a good overview of potential solutions before coming to a conclusion. In this way the IGDT acted as a true aid to the designer in finding the most satisfying solution.

Acknowledgements

I wish to thank Dr. Werner Sobek, director of the Institute for Lightweight Structures at the University of Stuttgart for his patient support, as well as the University of Tennessee for a grant that aided in this work.

References

Bäck; Hoffmeister and Schwefel. "A Survey of Evolution Strategies", in: *Proceedings of the Forth International Conference on Genetic Algorithms*. Morgan Kaufmann Publ., San Mateo, California, 1992.Dawkins, Richard. *The Blind Watchmaker*. Harlow Logman. 1986.

Eshelman, Larry. "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.

Frazer, John. "The Dynamic Evolution of Designs". in: *4D Dynamics Conference on Design & Research Methodologies for Dynamic Form*. De Montfort University, Leicester. 1996.

Goldberg, David E. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley, Reading, Mass, 1994.

Gordon, William J.J. *Synectics: the Development of Creative Capacity*. Harper & Row, New York, 1961.

Holland, John H. *Adaptation in Natural and Artificial Systems*. The Univ. of Mich. Press, 1975.

Kelly, Kevin. *Out of Control: The New Biology of Machines, Social Systems and the Economic World*. Addison-Wesley, Reading, Mass, 1995.

Mitchell, A.G.M. "The Limits of Economy of Material in Frame-structures", in: *Philosophical Magazine and Journal of Science*. Volume VIII, Sixth Series, July - December 1904. Taylor and Francis, London, 1904.

Prince, George M. *The Practice of Creativity: a Manual for Dynamic Group Problem Solving*. Harper & Row, New York, 1970.

Purcell, T. A. and Gero, J. S. "Design and other types of fixation", in: *Design Studies* 17(4): 1996. pp. 363-383.

Reiser, A. Albert Einstein. Thornton Butterworth. London. 1931.