Scientific Research Publishing

American Journal of Computational Mathematics, 2017, 7, 291-320
http://www.scirp.org/journal/ajcm
ISSN Online: 2161-1211
ISSN Print: 2161-1203

Topological Design via a Rule Based Genetic Optimization Algorithm

David Webb[1], Qian Liu[2], Wissam Alobaidi[2*], Eric Sandgren[2#]

[1]Axalta Coating Systems, Front Royal, USA
[2]Systems Engineering Department, Donaghey College of Engineering & Information Technology, University of Arkansas at Little Rock, Little Rock, USA
Email: *wmalobaidi@ualr.edu, #exsandgren@ualr.edu

How to cite this paper: Webb, D., Liu, Q., Alobaidi, W. and Sandgren, E. (2017) Topological Design via a Rule Based Genetic Optimization Algorithm. *American Journal of Computational Mathematics*, **7**, 291-320.
https://doi.org/10.4236/ajcm.2017.73023

Received: July 22, 2017
Accepted: August 23, 2017
Published: August 28, 2017

Abstract

A topological structural design approach is presented which is based upon the implementation of a two phase evolutionary optimization algorithm in conjunction with a finite element analysis code. The first phase utilizes a conventional genetic approach which performs a global search for the optimal design topology. Dual level material properties are specified within the genetic encoding and are applied to each individual element in the design mesh to represent either design material or a void. The second phase introduces a rule based refinement which allows for user design intent to accelerate the solution process and eliminate obvious design discrepancies resulting from the phase one search. A series of plate design problems are presented where the objective is to minimize the overall volume of the structure under predefined loading and constraint conditions. The constraints include both stress and deflection considerations where stress is calculated through the use of a commercial finite element package. The initial plate example incorporates a coarse mesh, but a gradual decrease in element size was employed for the remaining cases examined. Replacement of the phase one search with a set of randomly generated designs is demonstrated in order to form a greatly reduced design space which drastically increases the efficiency of the solution process. Comparison results are drawn between the conventional genetic algorithm and the two phase procedure.

Keywords

Topological Design, Structural Optimization, Genetic Optimization, Variable Material Design

1. Introduction

Structural optimization has evolved considerably over the years to the point where it now represents an important design tool. During this evolution, progress has been made on the analysis side, the optimization side and the coordination between analysis and optimization. From the very beginning, structural optimization was viewed as a computationally intensive task and much of the effort was placed on efficiency. The implementation of sensitivity analysis for generating gradient information for size and shape optimization is one aspect of this early work. A review of the history as well as progress in structural optimization up to the turn of the century is available from a number of sources [1] [2] [3] [4] and [5]. The push to extend structural optimization into the topological realm resulted in several interesting new approaches. Most notably, the work of Bendsoe [2] introduced the concept of homogenization and material density variation. This work was important as the logical design approach is to consider an initial mass of material and to shape it in the optimal form for the design criteria imposed. Several alternative approaches were also introduced, building upon this concept, and taken as a whole; they define a design strategy based on the implementation of a genetic or evolutionary algorithm.

Topological optimization in conjunction with genetic algorithms has been implemented for the design of simple structures involving truss and beam elements [6] and [7] and also for plate elements [8] [9] and [10]. Topological optimization is closely linked to the conceptual design process as it does not require an initial design from which to work. A survey of other evolutionary algorithm applications in the structural design area is presented by Kicinger, *et al.* [7] and a demonstration of how a solution generated by a genetic algorithm is equivalent to one generated by a more traditional optimization approach is presented by Xie [8]. This traditional optimization approach presented by Xie [8] possesses limitations as solution convergence is exclusively dependent on the randomness of the genetic algorithm search. Presented within this research is a second phase approach to the generic algorithm through the injection of domain specific knowledge in the form of a rule based encoding scheme to enable increase solution convergence for the most optimum design. Evolutionary Structural Optimization (ESO) is an evolutionary method for exclusively removing elements through material property characterization which has been demonstrated by Yang, *et al.* [11] and a Bi-directional Structural Optimization version (BESO) of the approach which can add or delete elements has been developed [12]. Both approaches simulate design enhancement through element removal, however the BESO approach possesses the ability to also add elements to strengthen regions enabling more diverse design outcomes. While these design techniques are referred to as evolutionary as a design is modified over a period of iterations, they are not driven by a genetic or evolutionary optimization code. Finally, the concept of robust design has been utilized to drive a topological optimization by Kim [13]. The concept of a robust design is important as most available design techniques produce solutions which may be highly sensitive to small changes in design conditions or material properties.

The discretized nature of computational structural analysis provides a direct link to the encoding used in a genetic optimization algorithm. If each element is represented by an individual position in the chromosome, topological modification can be carried out by removing or adding individual elements. The removal of elements may easily be handled through the assignment of a weak material property which effectively eliminates the element, but does not require any re-meshing. Using this approach, a region, or allowable design space, can be identified and meshed into elements. The genetic algorithm will then select the elements required for the design topology which best satisfies the design criteria and constraints. Often, however, the result of an evolutionary structural optimization contains obvious flaws that an experienced designer would not allow. This is a result of the fact that a mathematical abstraction of the "real" design problem is being operated on by the optimization algorithm. In order to eliminate these design flaws, some knowledge of the structural design process must be embedded within the optimization process. Overall, the encoding or chromosome string of a traditional genetic algorithm is proportional to the complexity of the problem, however knowledge based encoding approach is independent of problem size and does not proportionally grow in complexity. A smaller knowledge based encoding scheme eliminates traditional genetic algorithm solution flaws, increases solution convergence time, and provides feedback to the end user regarding which key features of information were the most essential when deriving the final solution.

The fields of expert systems and neural networks may be utilized in order to capture elements of the manual design process. This may, however, limit the design to known techniques and procedures which removes the possibility of creating a novel or revolutionary design. The best of both worlds would be a process which exploits the benefits of a global optimization process, but is guided or influenced by domain specific knowledge. This is possible within the framework of a genetic algorithm. This process may well maintain the global nature of the search as well as to allow for an adaptive framework over time as rules are modified to improve performance. A simple framework is presented, but this framework provides a rich development platform for more sophisticated structural design optimization approaches.

A number of different encodings are possible with a genetic algorithm for topological design. In order to be a useful design tool, however, the desire is to let the algorithm have complete control over the topology of the design. This means the algorithm must be capable of deleting as well as adding elements or features to the design. For a plate design, only the loading and ground locations should be specified initially along with a feasible region of space in which the design must fit. Each analysis of a structure is time consuming which brings up the issue of efficiency. Design population size must be limited, which in turn limits the effectiveness of the approach. The traditional genetic algorithm exploits successful design content, but this is not the same as exploiting design knowledge. A two phase genetic algorithm for robust topological design is developed herein. A series of design problems utilizing plate elements is presented and the results of the two phase algorithm are contrasted to those generated by a traditional genetic algorithm. Only loading conditions, restraints and geometric boundaries are specified for each problem.

2. Background

The plate optimization procedure considered here operates through the assignment of different material properties throughout a predefined meshed design region. In order to simplify the concept, only two different material property values are introduced. For a plate design, the logical material property to consider is the modulus of elasticity. The first material property value represents the intended design material such as that for steel or aluminum. The second material property value corresponds to an extremely weak material which adds little to the structural integrity of the design. The utilization of a weak material allows for extreme topological change to occur without requiring a re-mesh for the elements and eliminates the problem of generating a singular stiffness matrix. The stiffness matrix for the finite element analysis of the design is assembled with material property information provided by the genetic encoding on an element by element basis. The final topology of the design may be identified by simply removing the elements formed of the weak material. This is not intended to provide a final, highly detailed design, but rather a topology which may be refined in order to generate such a design.

The plate design process investigated incorporates a genetic algorithm programmed in Microsoft Visual Basic [14], and CAEFEM [15], a finite element solver which performs a linear static analysis for each design topology considered. The plate design mesh is constructed using Femap [16], a three dimensional CAD package, which generates a neutral file that the finite element solver can read and evaluate. This neutral file [17] documents the geometry as well as all of the material properties and other parameters of each plate design investigated. The genetic algorithm generates designs through the alteration of the material property of each plate element within the selected neutral file. The alteration of material properties can generate a realm of diverse topological designs, where each design is processed by altering the neutral file and passing it on to the finite element solver. The element stresses and the nodal deflections are computed by the finite element solver and subsequently, this information is utilized to evaluate the objective function and constraints which are defined by the optimization formulation. The specific form of objective function in this application was chosen to be the minimization of the total volume of a plate based design subject to predefined loading and constraint conditions. Stress and displacement limits served as structural constraint bounds.

A major issue of concern in the assignment of an alterable material property for each element in the defined mesh is the total number of elements considered.

As the number of elements increases, the complexity as well as the time required for solution by the optimization algorithm increases significantly. On the other hand, the number of elements required to define the fundamental design topology will generally be considerably below that required for a detailed design analysis. Another approach is to limit the optimization to a specific region of a design, where the topology of other regions is already fully defined. Meshing at several levels of detail is considered in the examples which follow. This is somewhat equivalent to the refinement provided by a variable length encoding [13] which could be implemented in order to automate the overall process. The fundamental question which must be addressed is whether a rule based genetic algorithm is capable of solving a realistic structural design problem with a reasonable amount of computational effort. The process is inherently parallel in nature which could lead to significant reduction in computational time but not in computational effort. Extensions to problems involving three dimensional solid objects are straightforward, using the same design optimization strategy with an expanded genetic encoding.

3. Problem Formulation and Phase One

Two separate genetic optimization formulations are introduced. The first represents a traditional encoding, where each element in the defined structural region has its own binary variable in the encoding string. This formulation is developed in this section. The development of the rule based extension follows in the next section. The plate design examples investigated were all subjected to identical stress and displacement constraint conditions. Each plate design was also restricted to the same spatial design volume, although the discretization level or element size was varied. The dimensions of the spatial design volume consisted of eighteen cm in the x direction, thirteen cm in the y direction, and a thickness of 0.5 cm in the z direction. Directional forces consisting of 67 newtons in the x, y and z directions were applied at coordinate locations (15, 3.25, 0.5) and (12, 9.75, 0.5) where the units are in cm and the origin is at the lower left corner of the bottom surface of the specified plate design volume. The use of multidirectional forces allow for an optimal yet robust topological design to be acquired. The concept of robust design optimization has been effectively demonstrated by Sandgren and Cameron [18] for truss structures as well as an automotive inner body panel. Mesh sizes ranged from twenty four to two hundred thirty four elements. The maximum allowable stress constraint was set at 140 MPa and a maximum allowable displacement constraint was defined to be 0.635 cm. The objective function used to evaluate each design is based upon the amount of volume removed from the original mesh. This objective function value is equal to the total element volume of weak material present in each design as specified by the genetic algorithm encoding.

The formulation of the objective function and constraints for the plate design examples considered is as follows:

$$\text{Maximize volume} = \sum\left(\text{Volume}_i\right)*\left(\text{Encoding value}_i\right) \tag{1}$$

$i = 1, \cdots,$ number of elements

subject to

$$g_i\left(x\right) = \left(\frac{\text{Constra int}_{\lim it} - \text{Constra int}_{\max}\left(\text{Calculated}\right)}{\text{Constra int}_{\lim it}}\right) \geq 0 \tag{2}$$

where $i = 1, \cdots,$ number of constraints

$g_1\left(x\right)$ correlates to stress and $g_2\left(x\right)$ displacement

The encoding value for each element has a value of zero if the element is assigned to the design material and one if it is assigned to the weak design material. This way, the summation of volume to be maximized in Equation (1) consists of the total volume of weak material elements, or alternatively, the volume of material removed. In the constraint equations, S_{limit} and D_{limit} are the maximum allowable stress and displacement values while S_{max}(calculated) and D_{max} (calculated) are the maximum computed values for the design being analyzed over all of the elements considered. The formulation of Equation (2) generates ratios, which allow for normalization of constraint violation in the disparate magnitudes of the displacement and stress values as well as to allow for a consistent graphical representation over the diverse set of designs within each genetic population. Both constraint equations produce positive values for any design which does not exceed the design limits.

Material properties of each plate element were designated to be either the intended design material, AISI 4340 steel, or a substantially weaker material with a modulus of elasticity of 2 MPa. Additionally, Poisson's Ratio was assigned a value of 0.32 for all elements in each plate design investigated. The modulus for the weak material may be defined as any reasonably small value as long as it is significantly less than that of the selected design material. The main function of this material property is to prevent the formulation of a singular stiffness matrix within the finite element algorithm and to avoid the necessity of re-meshing the design region. The assumption is made that low stress magnitudes in plate elements assigned the weak material property are deemed to be practically nonexistent. Steel material properties were represented the by the encoding value of 0, while weak material properties are assigned the encoding value of 1. These assigned numeric values correspond to defined material properties generated within Femap which are encoded within the genetic algorithm for material property manipulation of each element. A sample string is shown below for a ten element plate.

Sample string {0, 1, 1, 0, 0, 0, 1, 0, 0, 1}

For this string, elements one, four, five, six, eight and nine are formed from steel, while elements two, three, seven and ten are formed from the weak material. The assignment of order in the encoding string is directly related to the element number in the design mesh. Since the design mesh remains constant throughout the optimization process, the encoding strategy is valid throughout the search.

Based upon the structure of the sample string, an initial set of designs are generated randomly and the genetic algorithm methodology begins. Both crossover and mutation operations are applied by the genetic algorithm, and ultimately new generations of offspring or designs are formed. The method of crossover involves the selection of two parent designs where material properties or chromosomes within each chosen design string are interchanged at a random position, forming two new offspring designs. Parent designs are selected randomly, based on their overall fitness value, which considers the value of the objective function as well as any constraint violation. A single numeric value for each design is formed through the use of an exterior penalty function. Using the penalty function allows parent design strings which meet or exceed predefined constraint values to be assigned an overall fitness value solely based upon the objective function value. Parent designs which fail to meet the constraint criteria, are reduced in value by a penalty function. A typical penalty function is represented by the equation

$$P(x) = f(x) - R\sum_i \{g_i(x)\}^2 \tag{3}$$

for $i = 1, \cdots,$ number of constraints
where

$$\{g_i(x)\} = 0;\ for\ g_i(x) \geq 0$$
$$\{g_i(x)\} = g_i(x);\ for\ g_i(x) < 0$$

In this equation, f(x) represents the objective function value at the design point x, which specifies how much material was removed from the spatial design region. The penalty factor, R is a numeric value which continually increases from an initial value by a user specified amount as the optimization proceeds. This allows an early exploration phase where some constraint violation is allowed, but builds a penalty over time so that later designs are pulled toward the feasible design region. Lastly, the array $g_i(x)$ represents the constraint value for each of the defined constraints. Additional background on penalty function theory and operation is provided by Gen and Cheng [19]. The penalty function or specific fitness value for each population member determines which members of the population are best suited for producing new design offspring, resulting in the term parent designs. An in depth discussion of parent selection as well as the genetic optimization process is given by Goldberg [20] and Davis [21].

Mutation is a random occurrence of an altered material property value within an arbitrarily selected parent design string which occurs during a crossover operation. This random alteration allows for the possibility of a solution to be generated which could not result from any combination of the encoding strings represented in the current population. Although a random occurrence, the probability of mutation is user specified and normally set at a relatively low value. If the mutation probability is set too high, the search becomes more of a random search rather than an ordered search. Toward the end of the search process, most progress is made through mutation as most of the design content in the original population of designs has been exploited or lost. In a computationally expensive problem environment, such as structural design, it is important to have a reasonable mutation rate as the population size must be limited.

Once the crossover of traits between parent designs has been completed, stresses and displacements for each of the offspring designs are calculated and evaluated in the stress and displacement constraint functions. Element stresses and nodal displacements are immediately available once the finite element analysis algorithm, CAEFEM, has completed the analysis. Now the evolutionary process has commenced. This evolutionary process begins with the selection of new parent designs from the current generation. Once all parent designs have bred new offspring, one generation has been completed. The total number of generations of offspring produced is a user specified variable, which is application specific. The effectiveness of a specified set of input parameter values may be observed through feedback from the graphical user interface.

4. Phase Two

The second phase of the optimization process is designed to refine the best design or designs from the phase one search through the implementation of domain specific knowledge provided by the user in the form of rules. These rules are created by the user and encoded into a design string similar to the element selection string within the phase one search. Upon the conclusion of phase one, a global search has been performed which should result in a reasonably good design, which is at or close to a feasible design topology. Since by nature, the genetic optimization process tends to make good progress early in the optimization, the location of a set of reasonable design points does not require many generations to be executed in the phase one search. Alternatively, the phase one search can be done away with entirely and replaced by a set of randomly generated designs. While this process may lead to a local minimum, depending upon the diversity of the randomly selected designs generated, it allows for large problems to be solved using the greatly reduced encoding length of the rule based search. The alternatives supported by the two phase design optimization methodology lead to a rich set of alternatives which can be implemented to balance computational effort with the scope of the global search.

The fundamental operation of the genetic algorithm remains identical to the phase one operation, with the exception of the development of rule strings which are mated and offspring rule sequences which are introduced to further refine the final solution(s) from the phase one search. Each time the genetic algorithm has located a rule sequence for which both stress and displacement constraints are satisfied and an improvement in the objective function is achieved, the improved design replaces the current design. This rule based refinement is based upon the single row encoding scheme illustrated in **Figure 1**.

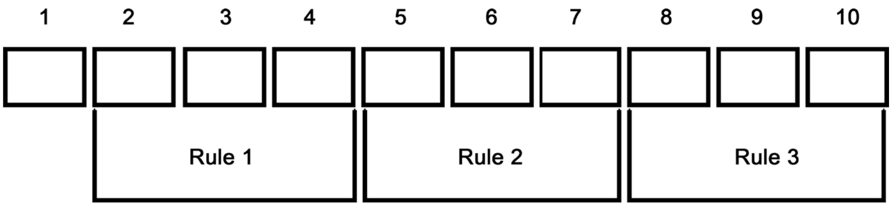

Figure 1. Phase two operation layout.

This specific encoding example consists of ten columns, where each column or block position contains information that is necessary in order to execute one or more of the rules. This particular encoding method allows for up to three rules to be executed simultaneously. The first position in the encoding string determines how many rules are to be executed in order to modify the current design. This leaves three groups of three block positions to provide the information necessary to execute specific rules. The first of the three block position values determines which rule to execute. The second and third block position values provide any additional information necessary to execute the selected rule (*i.e.* specific element to manipulate). This encoding method is repeated throughout the remaining two block sets which compose the remainder of the encoded row. This encoding methodology provides the foundation for the development of design strings which are injected into the genetic algorithm to refine the best design(s) from phase one. Phase two has the ultimate goal of determining the best rule sequence, from a global perspective, which refines a previously located design point from the phase one search or the current design(s) in the phase two search.

Five rules were developed for demonstrating the operation of the phase two search process. Each rule works independently or in a synchronized manner with multiple rules in order to achieve an improvement in the objective function which is to reduce the overall volume of the current plate design. Listed below are the five rules developed.

1) Rule one indicates that for a randomly selected element, the material property is switched from its current material property to its opposite (*i.e.* weak material property to the design material property or vice versa).

2) Rule two switches the material properties of two randomly selected plate elements with one another.

3) Rule three locates the element of maximum stress. Upon the location of the element of maximum stress, if applicable, the optimizer alters the weak material property of the element to the design material property.

4) Rule four locates the element of maximum displacement, and alters the material property of the element to the design material property if the current material property was designated a weak material.

5) Rule five locates the element of minimum stress and switches the material property to the weak material property if the design material property was initially present.

Each of the rules has a unique assignment of the rule block positions which provides the information required to process the rules for a given topological plate design. In all cases, the first of the three block positions represents the rule number, which is an integer from one to five. The second block position is an integer representing the element number selected for modification (when required) and the third block position is only applicable for rule number two where it is an integer identifying the second element number selected for rule execution.

In order to clarify the interpretation of the rule encoding, consider the following string.

$$\text{Sample string} = \{3,2,12,56,5,21,13,1,33,44\}$$

As the first position has a value of three, all three rules represented by the string will be executed on the current plate design. The next three blocks in the encoding designate that rule two is to be executed which will exchange the material properties of elements twelve and fifty-six. The next block of three values indicates that rule five will be executed which relies on stress information from the finite element analysis (element of maximum stress) and therefore does not require the values in the following two fields (21 and 13). The material property of the element which has the maximum stress value will be altered to the design material if it is not already designated as such. The final three fields will execute rule 1 which will alter the material property of the thirty third elements to the value opposite the current setting. The last position, 44, is not required in order to execute this rule. The three modifications are made to the design and then it is evaluated for objective function and constraint values. If the design is improved, it is saved for future consideration by the genetic algorithm.

Other rules than those specified may be more effective in the phase two process, but the algorithm will adapt through the genetic process to locate the best rules and combination of rules. The rules are controlled by the genetic algorithm which in turn is utilized to modify a previously specified plate design. It should be noted that the rules are not required to be good design rules. For example, rule number four may be seen to be of limited value since the reduction of displacement at a nodal position will be influenced by all elements along the load path and particularly, those elements close to a ground position. The beauty of the rule based process is that good rules will be executed and bad rules will be avoided as the search progresses. At the end of the search, the user can see which rules or combination of rules lead to design improvements during the search. This can lead to rule refinement or even the discovery of a new design strategy. This links the process closely with learning and memory which are fundamental to any successful design process.

5. Results

The plate design examples differ only in the number of elements involved in the meshing of the design space. The specific input parameters utilized for each of the design exercises including population size and number of generations of

Table 1. Overview of population and generation sizes assigned to each plate considered.

Number of Plate Elements	Phase One Population Size	Phase One Generations	Phase Two Population Size	Phase Two Generations
24	48	24	100	40
88	176	143	100	40
140	280	140	100	40
234	468	420	100	40

offspring utilized for both the phase one and phase two search processes are documented in Table 1. From this table, it is noted that both the population size as well as the number of generations required increased as the number of elements increased in the phase one search. These parameters for the phase two search remained constant, regardless of the number of plate elements contained in the design space mesh. This is a direct correlation to the length of the encoding string which increases for phase one as the number of elements increases, but remains fixed for the phase two search component. This once again shows the promise of the design process using a rule based approach. Regardless of the size of the problem, the phase two search remains the same, reduced size. The results for each of the four mesh densities will be reviewed in detail. The phase one search is equivalent to a traditional genetic optimization, while the two phase search procedure represents the rule based, global search process.

6. Example One

The first example utilized a twenty four element mesh to fill the designated design space with the predefined loading and constraint conditions established for all of the examples. The 0.5 cm. thick plate is fixed in all directions along the left edge, and two nodal forces were applied which consisted of sixty seven Newtons in the x, y, and z directions. A maximum stress constraint of 140 MPa and a maximum displacement constraint of 0.635 cm. were imposed as well. The phase one population size and number of generations produced were 48 and 24 respectively. The resulting plate design from the phase one search is illustrated in Figure 2. The dark shaded regions within Figure 2 represent the elements which were assigned the design material. The light regions indicate elements which were assigned the weak material and are treated as voids. The element numbering proceeds from left to right by rows, starting at the bottom row. The final objective function value for the design shown in Figure 2 was 67.25 cm^3. This represents the volume removed from the original design region.

The phase two search consisted of a population size and number of generations produced of 100 and 40 respectively. The phase two search refined the phase one design to that shown in Figure 3. This design can be seen to be a much “cleaner” design and contains no obvious design flaws other than the diagonal attachment of several elements. This problem could be resolved by adding appropriate

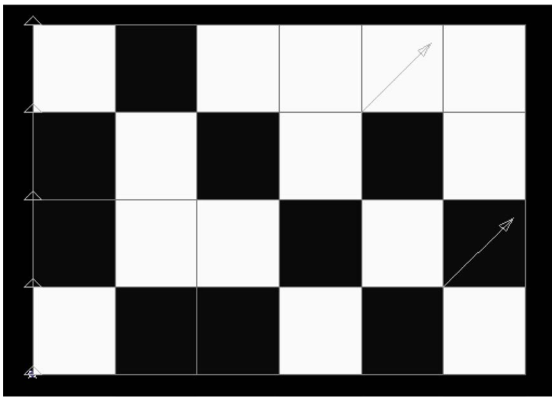

Figure 2. Phase one solution.

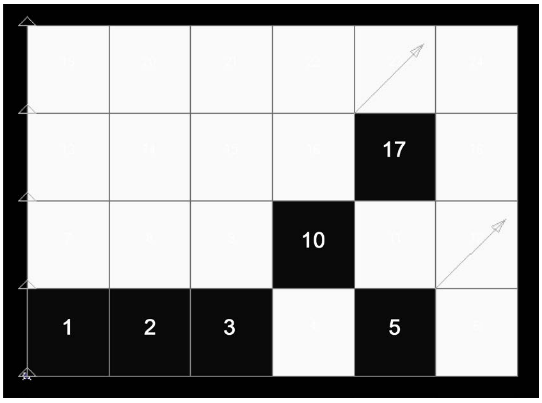

Figure 3. Phase two solution.

constraints or by other means [19] [20] [21] and [22]. This issue is not dealt with on the examples presented as only the design topology is desired and refinement of this topology is considered as a separate task. While the result pictured in Figure 2 from the phase one search contains elements that are beyond the point of force application, the phase two result does not contain such elements. The total volume removed for this design is improved to a total of 87.75 cm^3. The results remain rather crude due to the selected course mesh size and this leads to the question of whether the design would improve with a decreased mesh size.

Graphical representations for both displacement and stress versus element number are provided below in Figure 4 and Figure 5 upon the conclusion of phase two operation. The cyclical nature of the plots is a direct result of the element numbering scheme. The elements of highest stress and displacement are identified as element one and element five respectively. The maximum stress and displacement values fall slightly below the values allowed for the design. The large mesh size does not allow for the removal of any more material as eliminating one additional element would have lead to a design which violated the constraints. This once again is an indication that further improvement might be possible with a finer mesh.

Plots of both the objective function and constraint values versus generation graphs are provided in Figure 6 and Figure 7 which represent the genetic algo-

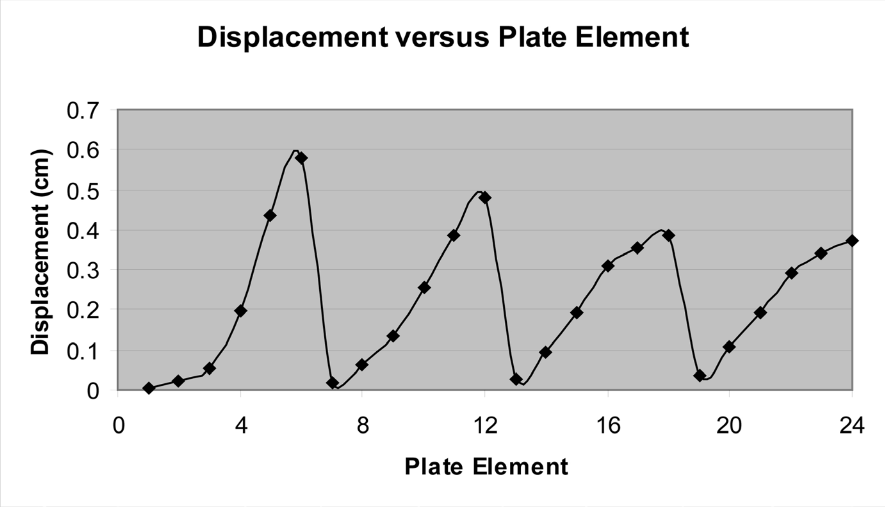

Figure 4. Displacement versus plate element.

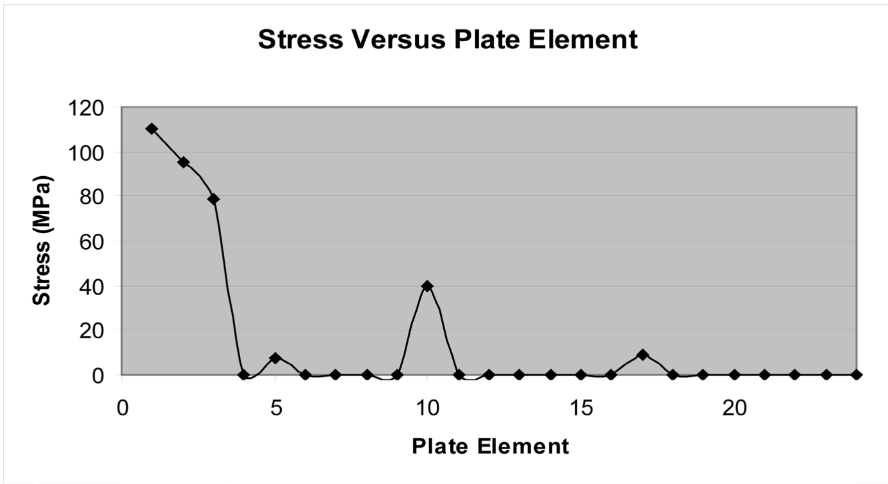

Figure 5. Stress versus plate element.

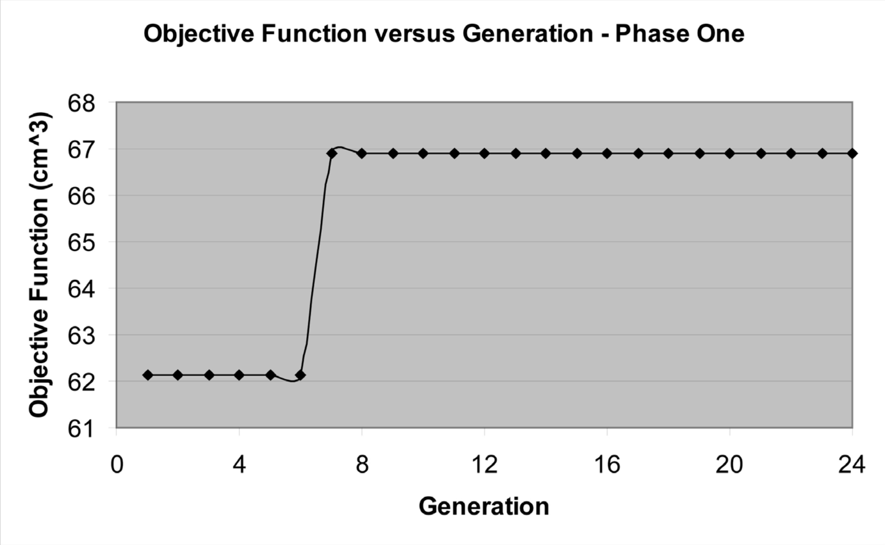

Figure 6. Objective function versus generation.

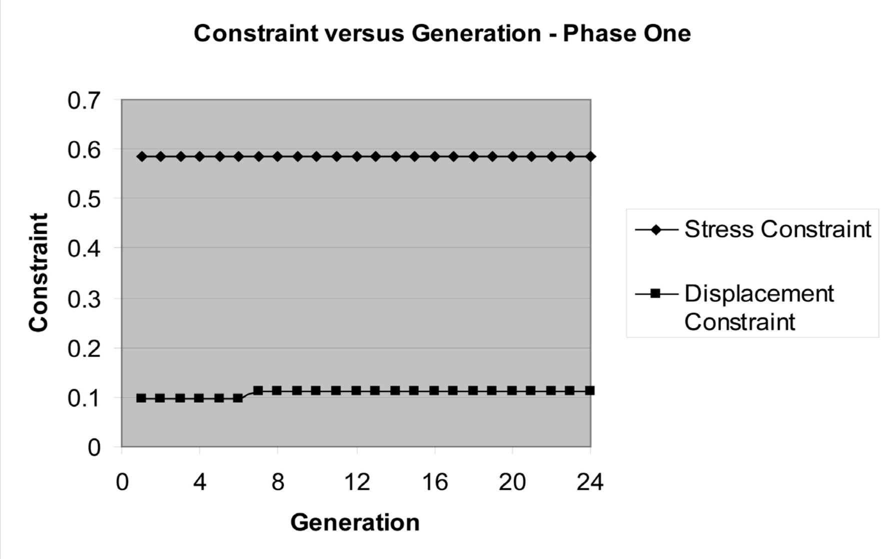

Figure 7. Constraint versus generation.

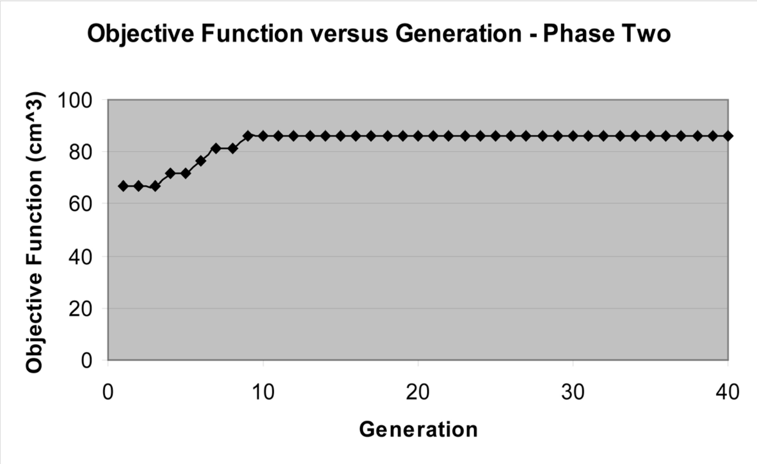

Figure 8. Objective function versus generation.

rithm behavior for the phase one search process. These graphical representations correspond to the final solution illustrated in Figure 2. Figure 6 indicates that the solution was located by the eighth generation and Figure 7 indicates that both the displacement and stress constraints were satisfied. The displacement constraint can be seen to be near its prescribed bound, while the stress constraint is far from being active.

Figure 8 and Figure 9 provide the solution history for the phase two search process. Figure 8 contains a plot of the objective function for each generation of designs produced and from this plot it can be seen that little improvement in the phase one design is produced after the tenth generation. The resulting objective function from the phase two search was enhanced from 68.25 cm^3 to 87.75 cm^3 which resulted in an additional thirty percent increase in the amount of material removed from the design. The overall stress and displacement values, shown in Figure 9 document that the phase two solution lies closer to the stress constraint

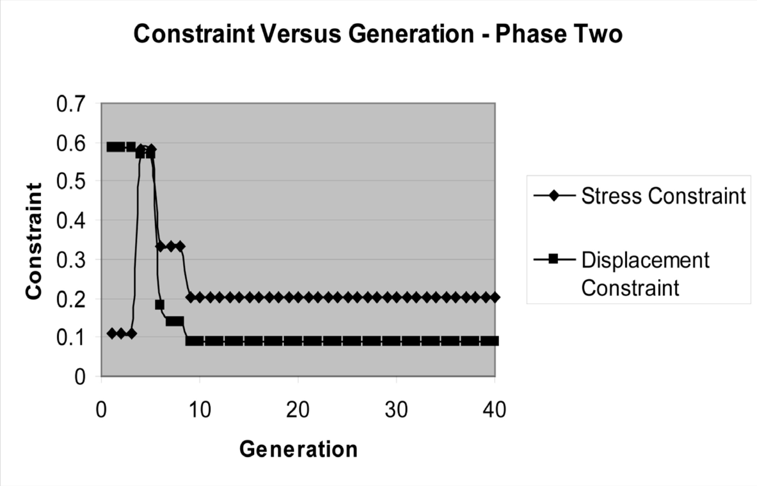

Figure 9. Constraint versus Generation.

limit compared to the phase one solution.

7. Example 2

The first decrease in the mesh size for the prescribed design region resulted in an increase in the number of plate elements from twenty-four to eighty-eight. All other design parameters remained at identical levels as in the previous design example. Solutions for both the phase one and phase two search are provided below in Figure 10 and Figure 11.

An examination of Figure 10 compared to the phase one solution for the course mesh from Figure 2 shows a significant change in the design topology. The objective function value was improved from the previous phase one solution of 68.25 cm^3 to a value of 93.07 cm^3. A more refined design was located upon the conclusion of the phase two search which is illustrated in Figure 11. The phase two search solution improved the objective function value to 97.06 cm^3 and removed extraneous elements from the phase one solution. The element identification numbers are provided for the elements formed from the actual design material in Figure 11, which correlate to the graphical representations for the element stress and displacement values.

The displacement and stress values for each plate element upon the conclusion of the phase two search process are presented graphically in Figure 12 and Figure 13. The nodal numbering scheme which is determined by the mesh generator makes it more difficult to identify specific element values, but it is clear from these graphs that both stress and displacement values are below the specified design limits. This once again points toward the possibility of a reduced mesh size allowing for even more material removal.

The progress of the genetic algorithm throughout the phase one search process is provided below through graphical representations of objective function value and constraint values versus generation, as shown in Figure 14 and Figure 15. The search did not converge until over one hundred generations had been produced and the displacement constraint was active at the phase one design solution.

The phase two optimization progress is plotted in Figure 16 and Figure 17. The phase two process converged in approximately ten generations and while

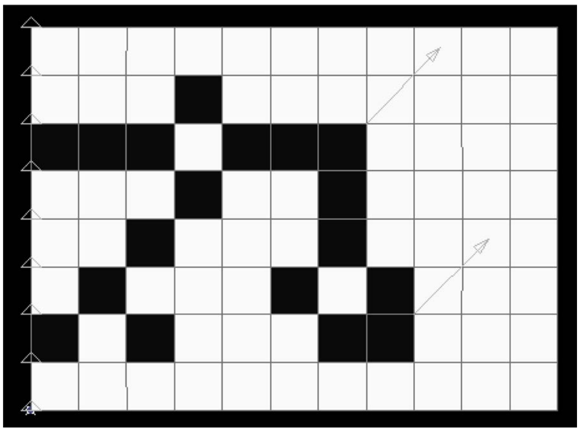

Figure 10. Phase one solution.

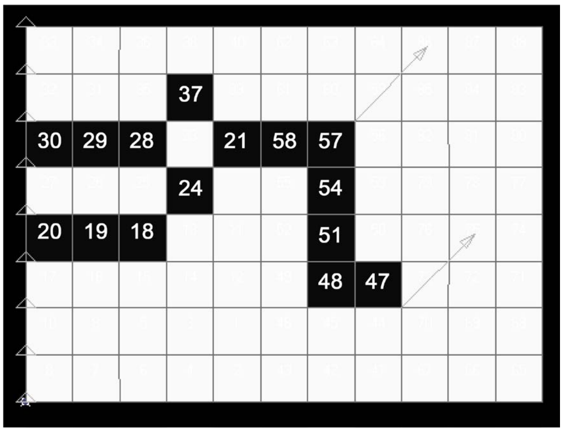

Figure 11. Phase two solution.

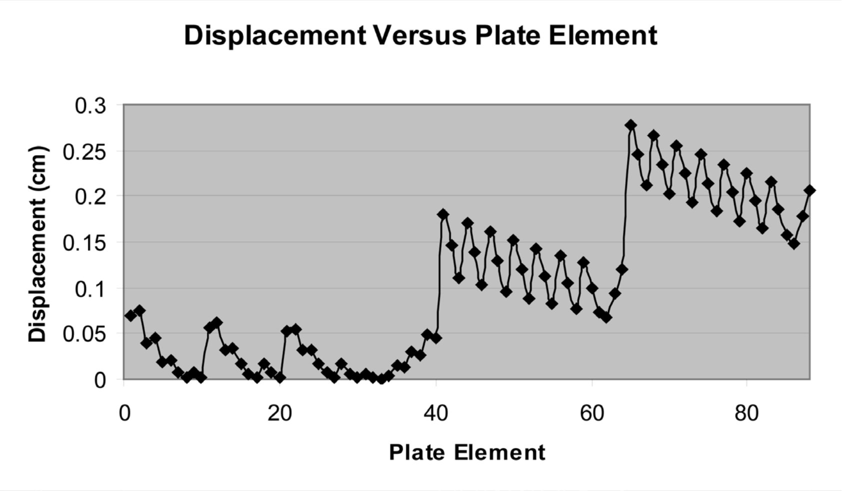

Figure 12. Displacement versus plate element.

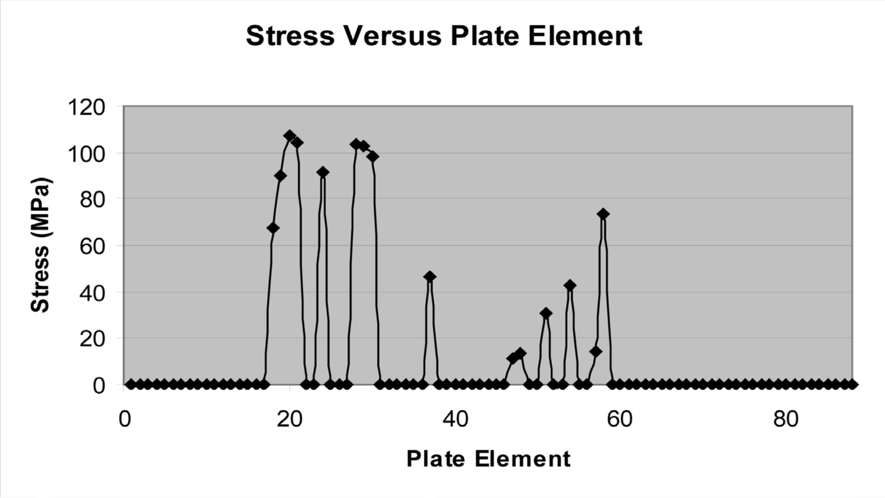

Figure 13. Stress versus plate element.

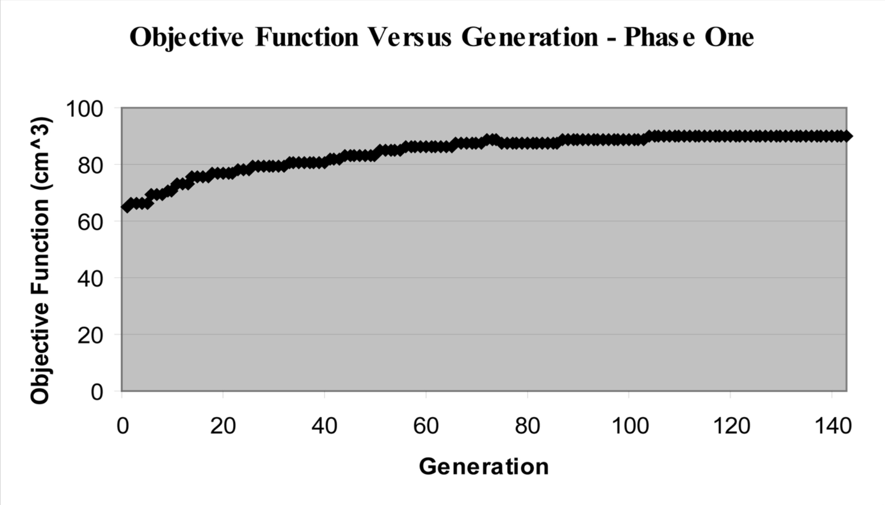

Figure 14. Objective function versus generation.

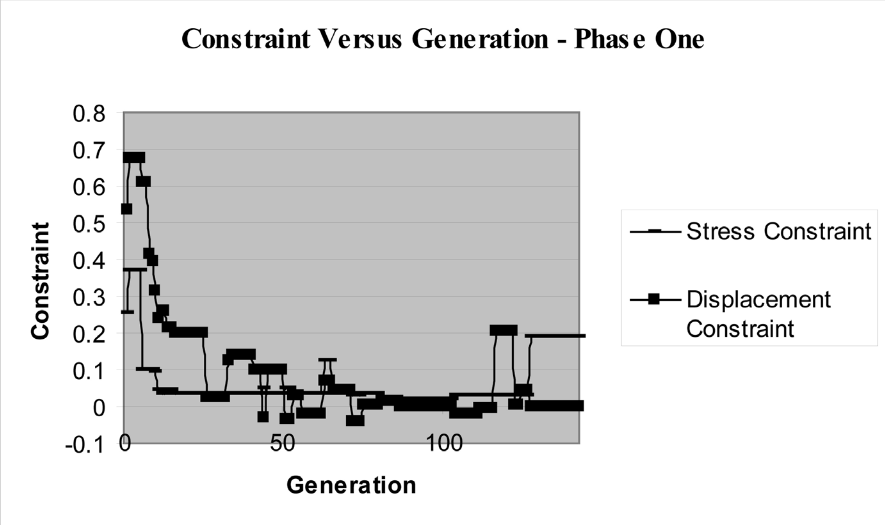

Figure 15. Constraint versus generation.

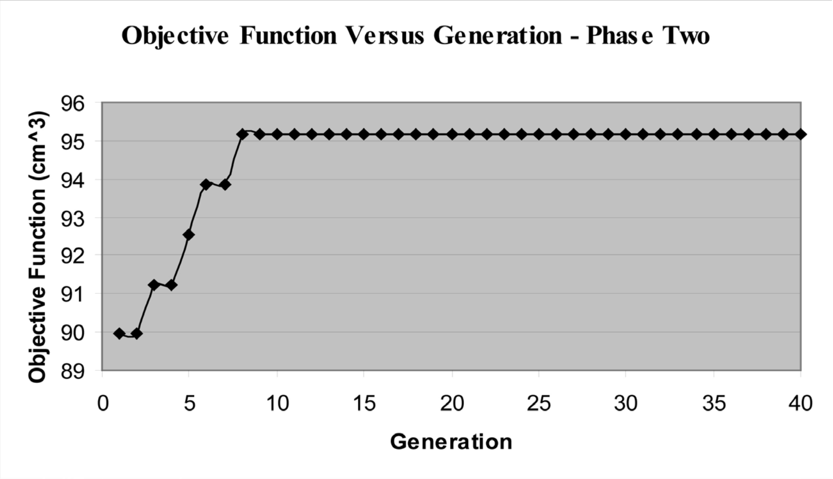

Figure 16. Objective function versus generation.

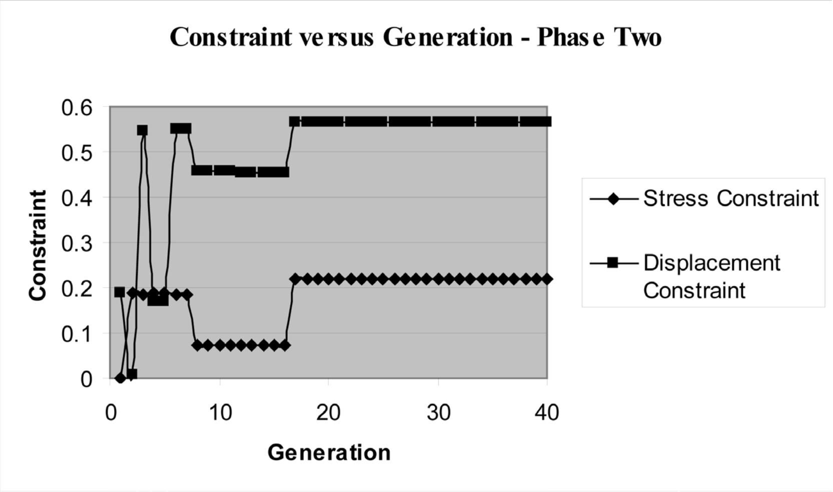

Figure 17. Constraint versus generation.

neither the stress nor displacement constraints were fully active, the design stress is approaching the specified limiting value. In this design exercise, the phase two search resulted in a more modest reduction of additional material, but the resulting design can be seen to be more refined in nature. The other interesting point of note is that the phase two result reduced the amount of material required for the design while increasing the safety margin of the design with respect to both displacement and stress criteria.

8. Example 3

Based upon the results achieved with the twenty-four and eighty-eight element plate design examples, it seems like a reasonable assumption that additional mesh reduction may allow for a further increase in the volume of material removed from the original design region. A one hundred forty element plate was constructed to investigate this possibility. Design topologies corresponding to the phases one and two results are illustrated in Figure 18 and Figure 19. An

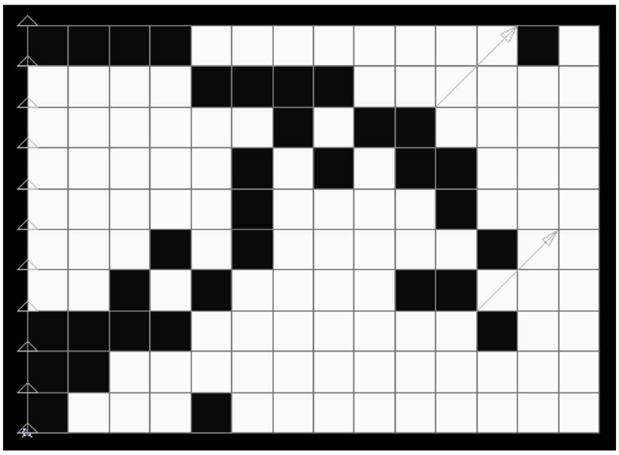

Figure 18. Phase one solution.

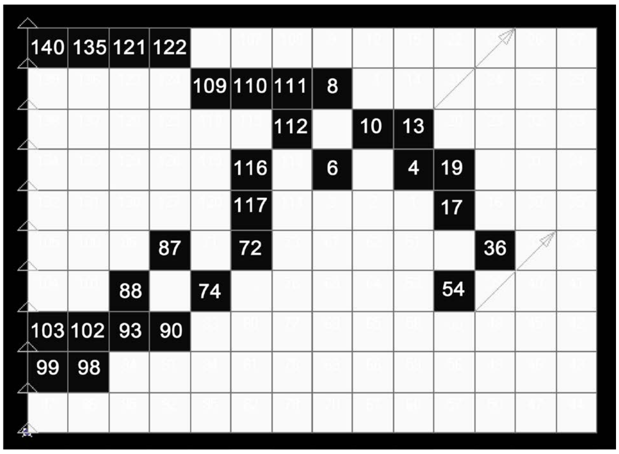

Figure 19. Phase two solution.

examination of Figure 18 reveals that the phase one global search solution had several disassociated elements in the final design as well as elements beyond the load application points which are both indicators of the level of difficulty this size problem presents for a traditional genetic algorithm in locating a refined solution. Once again, the phase two solution remedies the design discrepancies and produces a fairly refined design.

Graphical representations of displacement and stress values for each plate element in the final design after the phase two operation are provided in Figure 20 and Figure 21. From these plots it can be seen that both constraints are satisfied, although both are approaching the design limits imposed on the optimization formulation.

The behavior of the genetic algorithm is presented graphically for the phase one solution in Figure 22 and Figure 23. An objective function value of 88.59 cm^3 was located after approximately one hundred twenty generations of phase one operation as illustrated in Figure 22. This represents an interesting result as this is less material than for the phase one solution for the eighty-eight element design previously executed. This may be an indicator that the population size

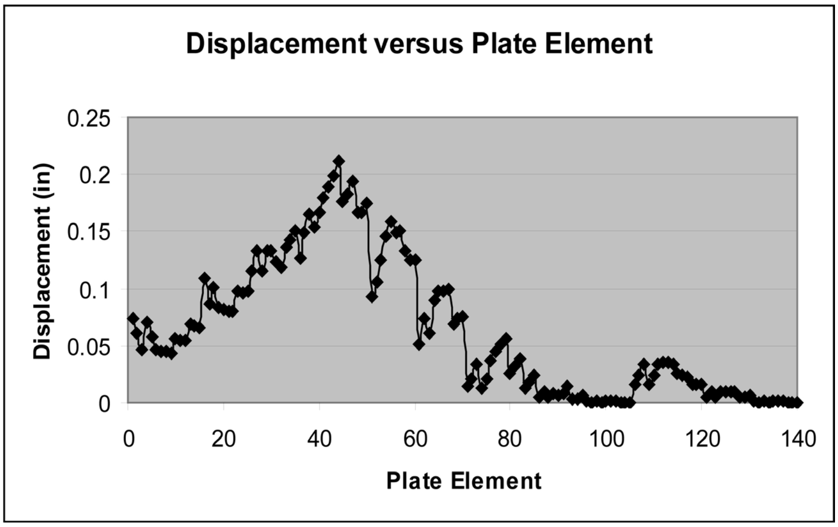

Figure 20. Displacement versus plate element.

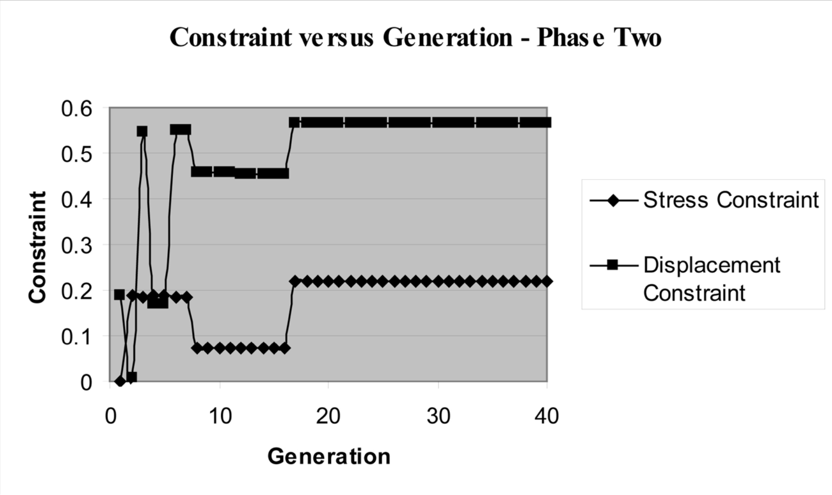

Figure 21. Stress versus plate element.

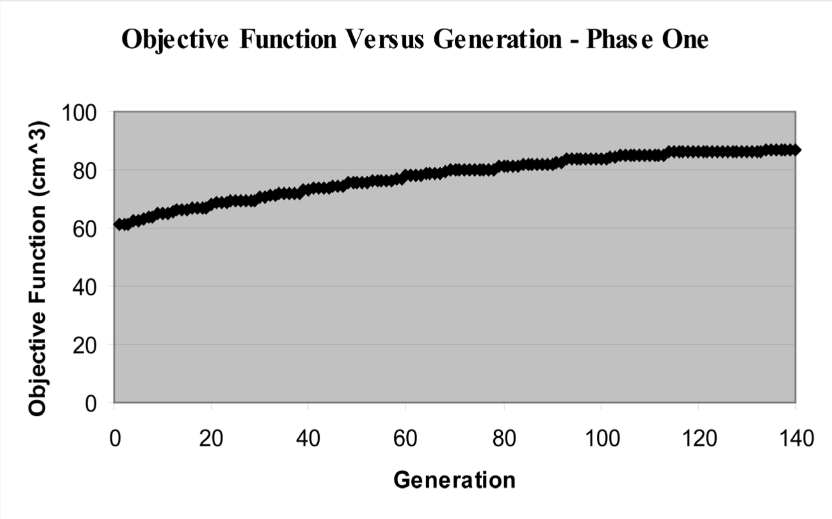

Figure 22. Objective function versus generation.

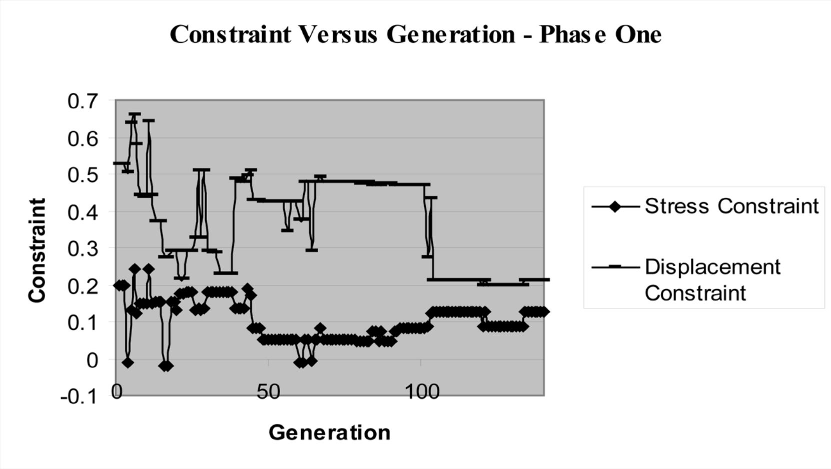

Figure 23. Constraint versus generation.

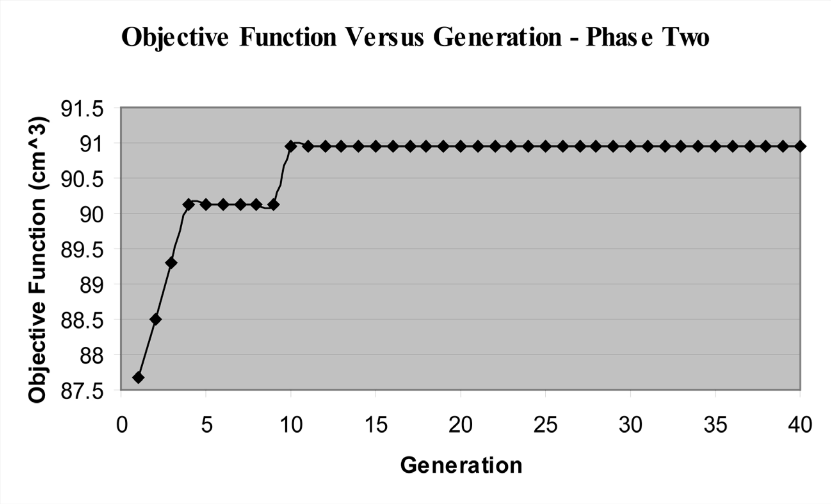

Figure 24. Objective function versus generation.

needs to be increased, but this would have a dramatic impact upon the solution time required. The constraint value versus generation graph displayed in Figure 23 reveals that the final phase one design satisfied both the stress and displacement constraint criterion.

Graphical representations of the phase two search process are illustrated within Figure 24 and Figure 25. Only ten generations were required to reach the final design solution. The objective function value was increased from 88.59 cm^3 to 92.76 cm^3 while maintaining acceptable stress and displacement design criterion. Once again, the volume of material removed is slightly worse than for the previous example. Both displacement and stress constraints are shown to be satisfied throughout the phase two process in Figure 25.

9. Example 4

The final plate design exercise investigated is composed of two hundred thirty four plate elements which are subjected to the previously defined loading and

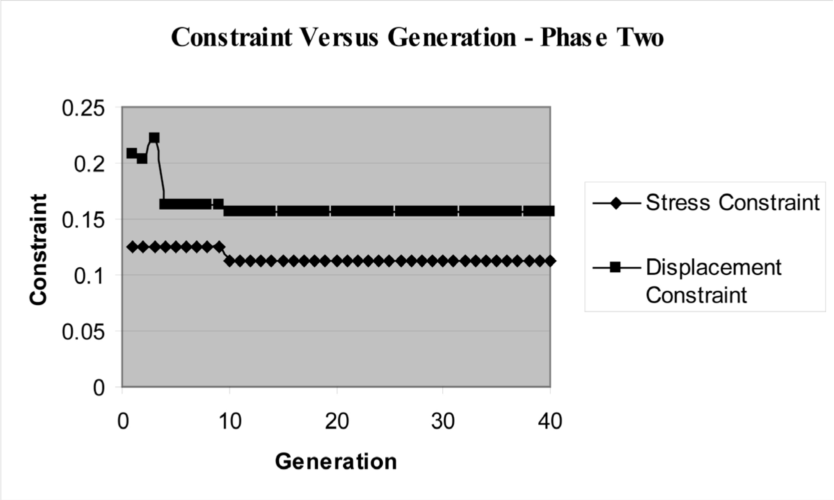

Figure 25. Constraint versus generation.

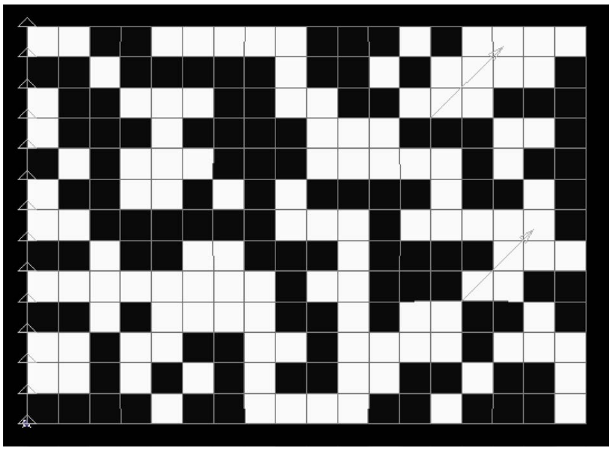

Figure 26. Phase one solution.

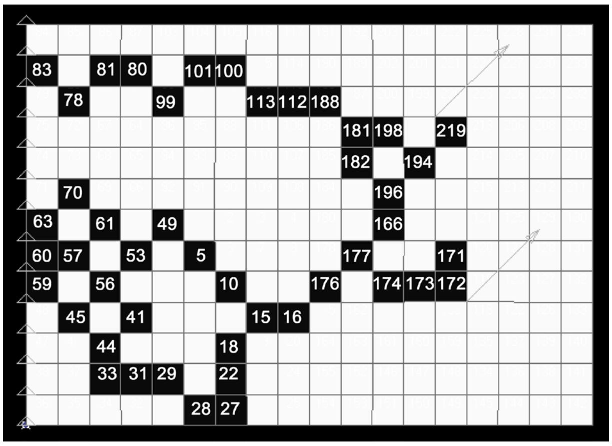

Figure 27. Phase two solution.

constraint conditions. Phase one genetic input parameters consisted of population size and number of generations of 486 and 420 respectively. Figure 26 and Figure 27 display both the phase one and phase two genetic optimization solutions. The phase one result is difficult to interpret as the final topology is not as well defined as for the previous plate examples considered. This fact is attributed to the substantial increase in the number of plate elements in this example which magnifies the difficulty in generating a solution through the use of a traditional genetic algorithm. Conversely, the phase two design shown in Figure 27 represents reasonable design solution, where elements are reconfigured and connected with surrounding elements and are no longer located beyond the defined nodal loading conditions. This is perhaps the best indicator of the power of the phase two approach as significant redesign was necessary in order to refine the phase one solution.

Displacement and stress values versus plate element number are presented graphically in Figure 28 and Figure 29. Both the displacement constraint and the stress constraint are seen to be active.

Objective function and constraint values versus generation graphs are provided below which plot the progress of the phase one genetic optimization

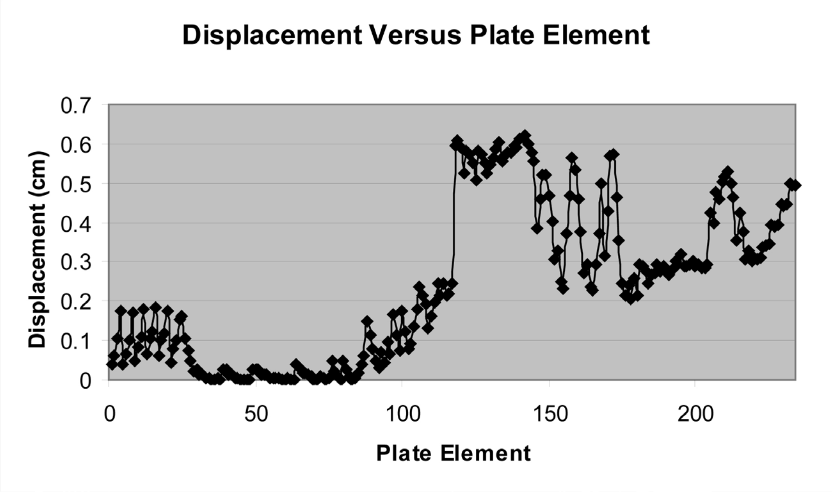

Figure 28. Displacement versus plate element.

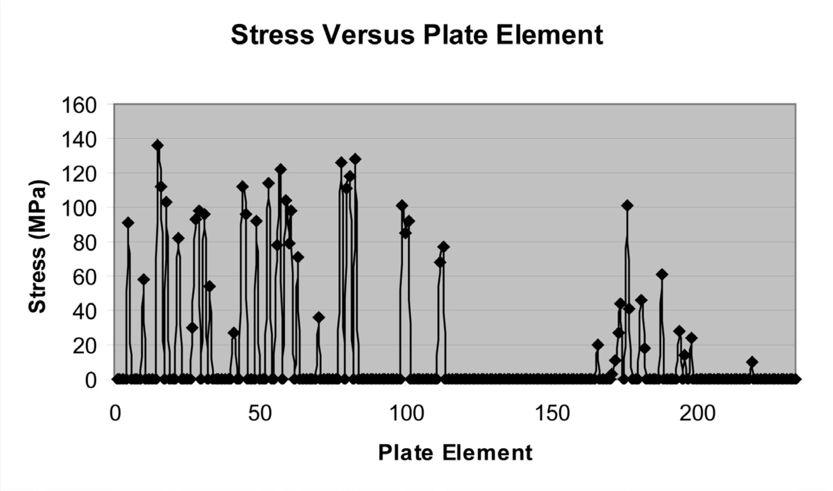

Figure 29. Stress versus plate element.

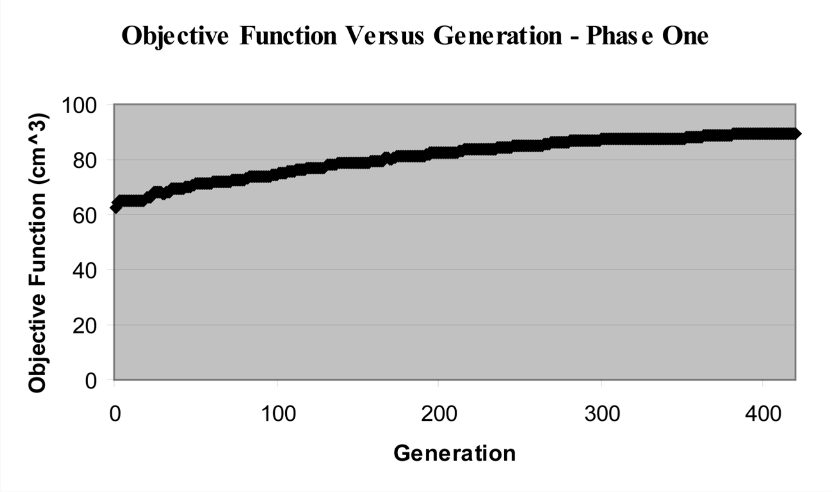

Figure 30. Objective function versus generation.

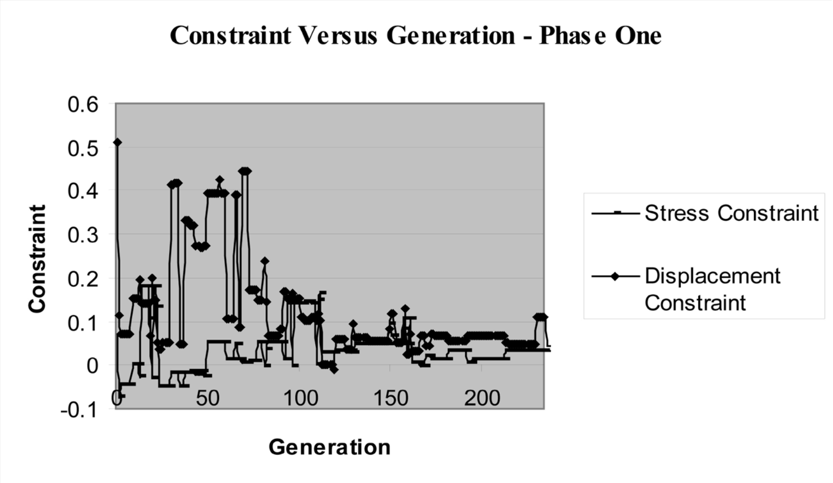

Figure 31. Constraint versus generation.

search. A steady increase in the objective function value for the phase one search is noted in Figure 30; however converged constraint values are not as evident within Figure 31, even after 400 generations. Additional generations might continue to slowly improve the objective function. The final objective function value attained upon the conclusion of the phase one search was 60.00 cm^3. This is significantly below the values reported previously for the larger mesh sizes and once again demonstrates the traditional phase one genetic search process is not adequate to solve a problem with this number of elements with a limited population size. From Figure 31 it is noted that both stress and displacement constraints are active at the conclusion of the phase one search.

Figure 32 and Figure 33 below display the phase two genetic behavior for both objective function value and constraint values versus generation. The objective function was increased to a value of 94.00 cm^3 in less than ten generations of offspring. This value is comparable with the best results achieved with the courser mesh sizes and is particularly significant in light of the poor progress

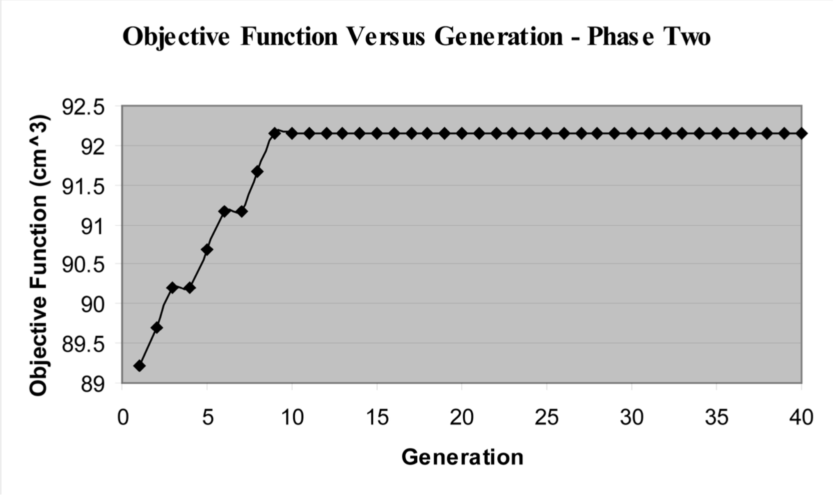

Figure 32. Objective function versus generation.

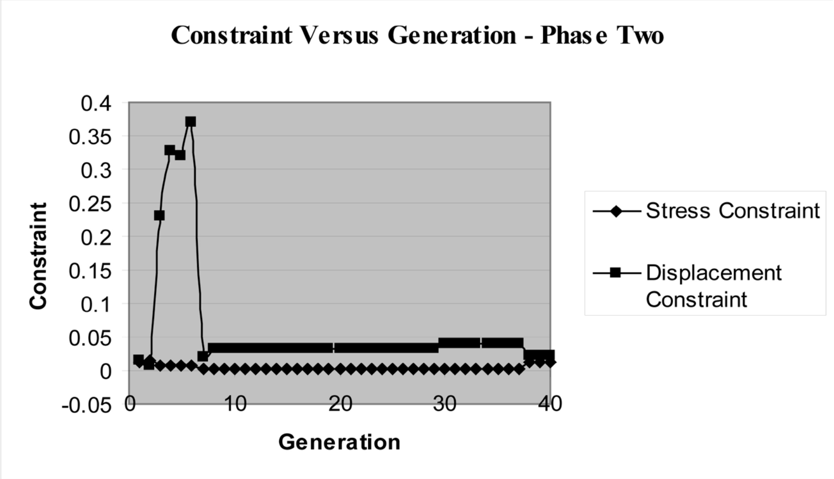

Figure 33. Constraint versus generation.

made during the phase one search process. Additionally, the constraint versus generation graph illustrated in Figure 33, is clearly indicates that both design constraints are active at the solution.

10. Extensions and Further Insight

The results presented to this point clearly point out the limitations of the phase one or conventional application of a genetic algorithm in setting the material properties for the topological optimization. The quality of the solution generated was reduced considerably at each mesh refinement stage and in the last case involving two hundred and thirty four elements, very little useful topology detail is available after the phase on e search process. The rule based, phase two implementation is remarkably more efficient and is capable of developing fairly refined topologies in within a very limited number of generations. This is due to several factors, the main two being the ability to infuse problem specific knowledge into the search and the reduction in the size of the solution space in the problem formulation. This brings up the issue of how much, if any, of a phase one search is required in the topological optimization. The other issue which deserves some additional consideration is the post evaluation of the rule usage during the phase two search process. Both of these issues will be addressed briefly.

In order to test the algorithm with a limited phase one search component, the 88 element plate problem was revisited. A phase one search was conducted with an extremely limited population size, 25, and for only 25 generations. The best final population member was then passed directly to the phase two search process which utilized a population size of 100 rule strings for a total of 40 generations. The final topology from the phase two search is presented in **Figure 34**. The final amount of material removed was 98.38 cm^3. This is actually slightly higher than the final two phase solution presented in Example 2, although the final topology is very similar to that presented in **Figure 11**. The objective function value and the constraint values versus generations conducted in the phase two search process are documented in **Figure 35** and **Figure 36**. Thus by reducing the total phase one function and constraint evaluations by over a factor of 40, did not damage the value of the final result. The limit of the reduction of the phase one search would be to stop the phase one search after the first generation which would produce a set of randomly generated designs. Utilizing this approach, the design space, or encoding length in the genetic algorithm, for a mesh containing tens of thousands of elements would be on the order of ten or twenty elements. Solving this size problem repeatedly, is demonstrated to be and expected to be far more efficient with respect to the number of function and constraint evaluations.

The issue of which rules are implemented and which of those rules were successfully utilized in the solution process provides an important look at how the general topological optimization process can be improved over time. The distribution of rule utilization for the solution of the 88 element problem with the reduced phase one search is shown in **Figure 37**. It was mentioned in the early discussion in the selection of the rules that rule number four was not constructed

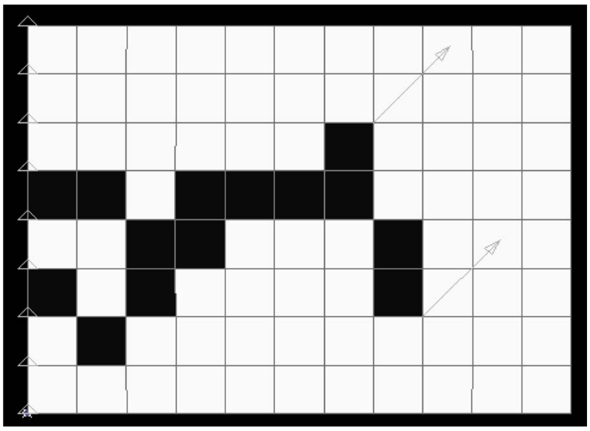

Figure 34. Final topology for plate design with reduced phase I search.

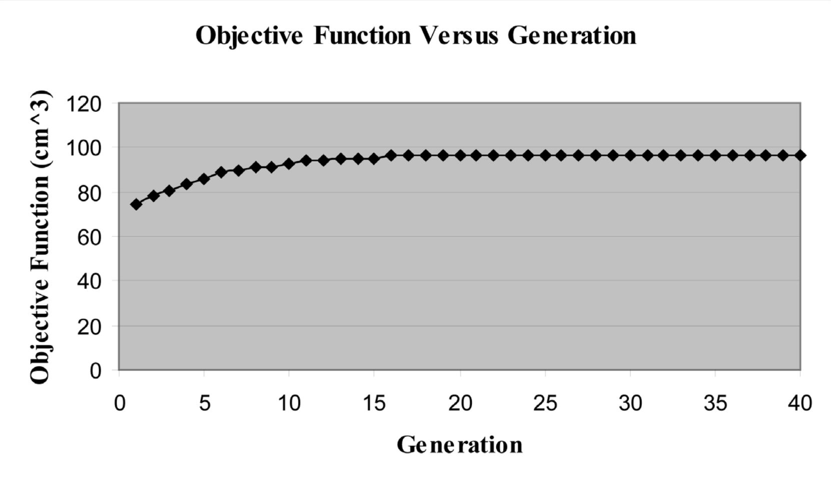

Figure 35. Objective function versus generation.

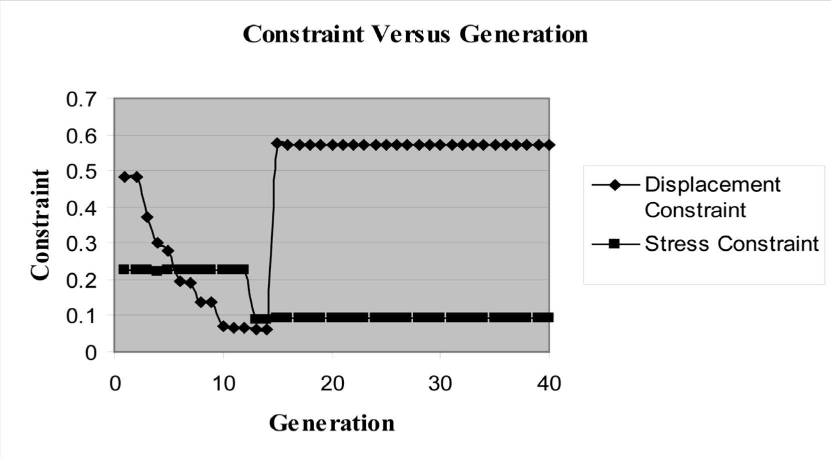

Figure 36. Constraint versus generation.

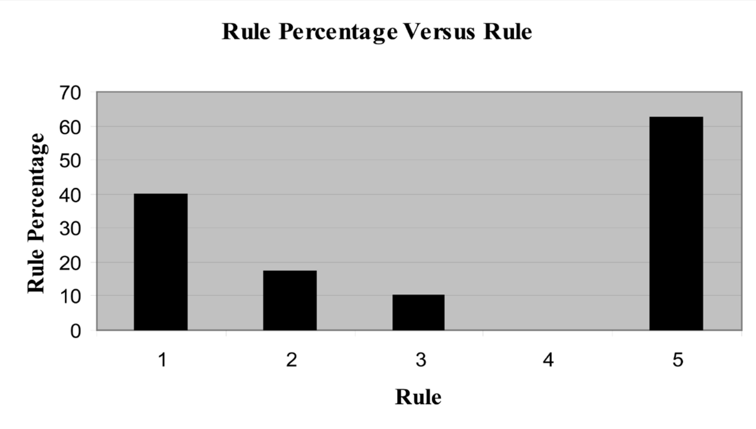

Figure 37. Rule Distribution for Successful Design Moves.

in a way that would improve the solution. This fact is documented in Figure 37 where it is indicated that the rule was never utilized successfully in the search. This information can be used to replace the rule with another, more appropriate rule. The distribution of the remaining four rules shows a dominance of rule number five, and lesser dependence on rules one, two and three. This means that for this particular case, the search was improved most often through the removal of material in regions of low stress. This is not an unexpected result, but the usefulness of building some randomness into the rule base is documented by the high rate of use of rules one and two. The interesting part of the rule development process over time, is that a better understanding of the optimization and design process is developed. This is a rare feature among any topological optimization procedure.

These insights to the rule development process and the ability to reduce or eliminate the phase one search process highlight the promise of the rule based approach. The implementation tested is one of a wide variety of similar algorithms which could have been constructed. Further understanding of the rule development process is needed as well as additional testing on larger topological design problems. The post process review of the successful rule distribution could be expanded to include rule interactions which would also provide useful design information. Based on the results achieved to date, however, the method promises to provide an effective design tool for structural design which provides insights into the solution process that are unavailable by most other approaches. The process represents a unique combination of an evolutionary search which is guided by domain specific knowledge.

11. Summary and Conclusions

The inclusion of domain specific knowledge in the form of design rules was implemented within the framework of a genetic algorithm and tested on a series of structural design problems involving plates. Each problem was subjected to identical boundary and loading conditions as well as the same original predefined design volume. The problems involved four different mesh sizes for the prescribed design region. The topology was defined by the assignment of material property values to each of the elements through the genetic encoding of the optimization algorithm. Two material values were assigned, a normal value for the intended design material and a value which represents a significantly weaker material which represents a void.

The results clearly demonstrate the lack of effectiveness of the traditional genetic algorithm implemented for the phase one search for this class of problems. The final designs generated by the phase one search are “dirty designs” as they contain obvious design flaws. The convergence of the phase one search was found to be slow which made it difficult to assign the population size required for a true global search to be performed. As the mesh was refined to form smaller elements, the problem size increased to the point where a reliable solution could not be located through the phase one search alone.

The addition of the second phase, rule based genetic algorithm was found to be extremely effective in cleaning up the design resulting from the phase one search. This process was found to require a very small number of generations, even when the number of elements assigned to the design region grew. The result from the phase one search was improved by the phase two search in all design cases considered. The issue of the topology generated as the number of elements increased was interesting. When relatively few elements were assigned, the final design was easily defined and clearly related to the design generated by the phase one search. As the number of elements increased, the final design topology represented a better design (less weight), but as the number was increased farther, the final design actually decreased in the measure of quality.

The level of difficulty certainly increases with the number of elements considered and this may partially explain this phenomena. Another possibility is that the topology is shifting from a global level to a microscopic level of detail. With large elements, the fundamental form of the structure is the major result. The approach clearly does not generate a final detailed design, but a clear topology is evident from which to form a final design. As the mesh is formed of smaller elements, the topology shifts to a finer grain which actually allows for more of a grain type structure to be formed such as that represented by a foam or matrix material containing small patterns of voids. Additional experimentation will be required in this area.

The two phase approach can be implemented in a host of different ways and the effectiveness of the resulting algorithm may improve in the process. Certainly, however, the initial results are promising. Completing the phase one search before implementing the phase two process may not be as efficient as a mixture of the phase one and phase two search where the phase two process operates on the entire phase one population. The development and representation of rules is also an area which requires further experimentation. The use of the rule history for improvement of the process is also an open area for further research. In any case, the basic methodology has been demonstrated and shown to be an effective approach to topological optimization. It certainly exceeds the performance of a traditional genetic based approach.

References

[1] Vanderplaats, G.N. (1997) Structural Optimization: Where We've Been and Where We're Going. In: Hernandez, S. and Brebbia, G.A., Eds., *Computer Aided Optimum Design of Structures V*, Computational Mechanics Publications, Berlin, 45-54.

[2] Bendsoe, M.P. and Kikuchi, N. (1988) Generating Optimal Topologies in Structural Design Using a Homogenization Method. *Computer Methods in Applied Mechanics and Engineering*, **71**, 197-224. https://doi.org/10.1016/0045-7825(88)90086-2

[3] Sandgren, E. (1997) Robust Shape Optimal Design with Consideration of Variation. In: Hernandez, S. and Brebbia, G.A., Eds., *Computer Aided Optimum Design of Structures V*, Computational Mechanics Publications, Berlin, 287-297.

[4] Alobaidi, W., Sandgren, E. and Alkuam, E. (2017) Decision Support through Intelligent Agent Based Simulation and Multiple Goal Based Evolutionary Optimization. *Intelligent Information Management*, **9**, 97-113. https://doi.org/10.4236/iim.2017.93005

[5] Alobaidi, W.M., Webb, D.J. and Sandgren, E. (2017) A Rule Based Evolutionary Optimization Approach for the Traveling Salesman Problem. *Intelligent Information Management*, **9**, 115-132. https://doi.org/10.4236/iim.2017.94006

[6] Sandgren, E. and Jensen, E. (1992) Automotive Structural Design Employing a Genetic Optimization Algorithm. *SAE Technical Paper Series*, Article ID: 920772.

[7] Kicinger, R., Arciszewski, T. and DeJong, K. (2005) Evolutionary Computation and Structural Design: A Survey of the State of the Art. *Computers and Structures*, **83**, 1943-1978. https://doi.org/10.1016/j.compstruc.2005.03.002

[8] Xie, Y.M. and Steven, G.P. (1997) Evolutionary Structural Optimization. Springer-Verlag, Berlin. https://doi.org/10.1007/978-1-4471-0985-3

[9] Eschenauer, H.A. and Olhoff, N. (2001) Topology Optimization of Continuum Structures: A Review. *Applied Mechanics Reviews*, **54**, 331-389. https://doi.org/10.1115/1.1388075

[10] Mackerle, J. (2003) Topology and Shape Optimization of Structures Using FEM and BEM—A Bibliography (1999-2001). *Finite Elements in Analysis and Design*, **39**, 243-253. https://doi.org/10.1016/S0168-874X(02)00071-9

[11] Yang, X.Y., Xie, Y.M., and Steven, G.P. (2005) Evolutionary Methods for Topology Optimization of Continuous Structures with Design Dependent Loads. *Computers and Structures*, **83**, 956-963. https://doi.org/10.1016/j.compstruc.2004.10.011

[12] Yang, X.Y. and Xie, Y.M. (1999) Bidirectional Evolutionary Method for Stiffness Optimization. *AIAA Journal*, **37**, 1483-1488. https://doi.org/10.2514/2.626

[13] Kim, I.Y. and de Weck, O.L. (2005) Variable Length Chromosome Length Genetic Algorithm for Progressive Refinement in Topology Optimization. *Structural and Multidisciplinary Optimization*, **29**, 445-456. https://doi.org/10.1007/s00158-004-0498-5

[14] Microsoft Corporation (1998) Microsoft Visual Basic Version 6.0, Microsoft Corporation, Washington DC.

[15] Concurrent Analysis Corporation (1998) Getting Started with CAEFEM Version 4.0, Concurrent Analysis Corporation, Encino.

[16] Femap (2002) http://www.eds.com/products/plm/unigraphics/ds/femap/

[17] Enterprise Software Products Inc. (1996) Femap Neutral File Format, Enterprise Software Products Inc.

[18] Sandgren, E. and Cameron, T.M. (2002) Robust Design Optimization of Structures through the Consideration of Variation. *Computers and Structures Publications*, 1605-1613. https://doi.org/10.1016/S0045-7949(02)00160-8

[19] Gen, M. and Cheng, R.W. (1997) Genetic Algorithms & Engineering Design. John Wiley & Sons Inc., New York.

[20] Goldberg, D.E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Boston.

[21] Davis, L. (1991) Handbook of Genetic Algorithms. Van Nostrand Reinhold, New York.

[22] Li, Q., Steven, G.P. and Xie, Y.M. (2004) A Simple Checkerboard Suppression Algorithm for Evolutionary Structural Optimization. *Structural and Multidisciplinary Optimization*, **22**, 230-239. https://doi.org/10.1007/s001580100140