

## Freeform Optimization: A New Capability to Perform Grid by Grid Shape Optimization of Structures

Juan Pablo Leiva

Vanderplaats Research and Development, Inc.  
41700 Gardenbrook, Ste. 115  
Novi, MI 48375, USA  
jp@vrand.com

### Abstract

This paper presents a new capability for performing a special type of shape optimization. Traditional shape optimization is often performed by using perturbation vectors that are linked with grid coordinates and design variables which an optimizer can change. In this work, these shape perturbation vectors are split so that each grid, and/or a set of nearby grids, and/or a set of grids linked by a manufacturing constraint, has its own design variable. This split produces great variability in the answers. In this work, possible distortions that could happen during an optimization run are prevented with automatically generated distortion constraints and/or by mesh smoothing. To distinguish this capability from standard shape optimization, we named this capability freeform optimization. This capability could be seen as a generalization of topography optimization since it can reproduce most of topography results, but it is different than topography optimization since it can be used to design any type of structure including solids and trusses whereas topography is mostly used for shell structures. Several examples which show the application of freeform are presented. One example will show the use of freeform optimization to find optimal rib locations in a solid structure. Other examples will demonstrate the use of freeform optimization to find optimal bead locations in shell structures.

**Keywords:** *Freeform, Shape, Topography, Optimization*

### 1. Introduction

Traditional shape optimization has been successfully utilized to optimize a variety of structures for many years. However, in today's competitive environment, where new designs are expected to perform better than previous ones and at the same time to be more economical, traditional shape optimization has started to be insufficient. This situation was similar for sizing optimization some years ago. Our response then, as software developers, was to improve on sizing optimization by developing topometry optimization [1]. Topometry optimization is essentially an automatically generated element-by-element sizing optimization. So when faced with the challenge to improve on shape optimization, we thought why not create something like a grid-coordinate by grid-coordinate optimization? Researching the literature we can find some papers on which each grid-coordinate is designed. The problem with this approach is that it can easily distort finite element meshes. In addition, this approach does not give enough controls to be useful. Because of that, we have been using topography optimization for shell structures and occasionally for solid structures. Topography optimization works by automatically creating perturbation vectors which in general are normal to the designable surface areas [2]. In Genesis, for shells, topography works directly [3]. For solids it needs to use an external shell skin that is used to design the surface grids of the structure and mesh smoothing to move the interior nodes. The use of topography optimization has given good results but has several limitations. The good results can be attributed to the fact that it uses perturbation vectors which are well defined. The most noticeable limitations are that: a) topography optimization can easily distort the mesh, b) topography only works with certain types of elements and c) topography can not produce beads with variable heights (user controlled maximum heights). Recently, we have developed a method to overcome the first limitation. The method uses internally created distortion constraints. Distortion constraints will be discussed later in this paper as they are new and very important to freeform optimization. Traditional shape optimization that uses perturbation vectors does not have the last two limitations described. In other words, it is not limited to any type of elements and perturbation vectors can take almost any shape we want. It should be mentioned here that there are several ways to create perturbation vectors and they can be created either internally in the software (for example using the Domain elements [4]) or using a design pre-processor like Design Studio [5]. A question arises: Can we take advantage of the best of topography (each grid has its own variable) and the best of shape optimization (perturbation vectors are known, have controlled shape and work with all elements)? The implantation of freeform is an attempt to answer that question. The idea behind freeform is the following: The program takes an existing perturbation vector and then breaks it into individual perturbations so that each grid has its own perturbation vector and its corresponding design variable. Because of this, we can define freeform optimization capability as a grid-by-grid shape optimization capability.

## 2. Approximation Concepts

Before going into the details on how freeform has been implemented, an explanation of how the Genesis program solves an optimization problem is presented next. In Genesis, a structural optimization problem is solved using the approximation concepts approach. In this approach, an approximate analysis model is created and optimized at each design cycle. The design solution of the approximate optimization is then used to update the finite element model, and a full system analysis is performed to create the next approximate analysis model. The sequence of design cycles continues until the approximate optimum design converges to the actual optimum design. In the mid-seventies Schmit et al. introduced approximation concepts for traditional structural optimization [5-6]. These concepts, in the eighties and early nineties, were refined to improve the quality of approximations [7-9]. The approximate problem is solved using either the BIGDOT [11] or DOT [10] optimizers. The purpose of using the approximation concepts approach is to reduce the number of design cycles to reduce time. With these approximations a good engineering answer can be typically found in 10 design cycles.

## 3 Shape Optimization

### 3.1 The Perturbation Vector

Shape optimization is used to find the optimal location of grids in a structure. In the Genesis software, shape optimization uses perturbation vectors. The perturbation vectors dictate how much and in which direction the coordinates of the grids can change when its corresponding design variable changes in one unit. The effect of one perturbation vector is shown in Fig.1 below.

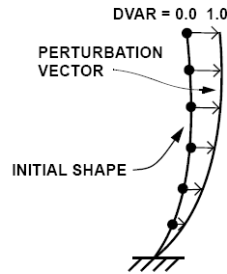


Figure 1. Perturbation Vector

The following figure shows how a perturbation vector affects the location of the grids. A value of 0.0 on the design variable causes no movement. A value of 1.0 causes the structure to move the same as the perturbation vector. If the design variables take a value larger than 1.0, then the grid movements are larger than the perturbation. In Fig. 2, below, it is shown how a shape would look if the design variable associated to the perturbation vectors took a value of 5.0.

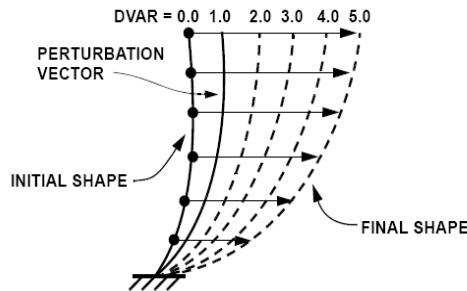


Figure 2. Shape Changes

### 3.2 Grid Location Update

When there are multiple perturbation vectors, the grid locations are updated by adding to the initial coordinate a linear combination of all perturbation vectors and its corresponding design variables. The equations used in shape optimization to internally calculate the new locations of the grids are:

$$\left. \begin{aligned} X_i &= X_{i0} + \sum_j DV_j * XP_{ij} \\ Y_i &= Y_{i0} + \sum_j DV_j * YP_{ij} \\ Z_i &= Z_{i0} + \sum_j DV_j * ZP_{ij} \end{aligned} \right\} \quad (1)$$

In equations (1),  $X_i$ ,  $Y_i$  and  $Z_i$  are the updated coordinates of the grid  $i$ .  $X_{i0}$ ,  $Y_{i0}$  and  $Z_{i0}$  are the initial coordinates of the grid  $i$ .  $XP_{ij}$ ,  $YP_{ij}$  and  $ZP_{ij}$  are the components of the  $j^{th}$  perturbation vector corresponding to grid  $i$ . Finally,  $DV_j$  is the value of design variable  $j$ . These equations are also used by topography and freeform optimization.

#### 4. Freeform Optimization

We can define freeform optimization as a grid-by-grid shape optimization capability in which a given perturbation vector is split to increase the variability of the answers. Besides the standard constraints, three special constraints can be used to improve the quality of answers: grid fraction constraints, distortion constraints and manufacturing constraints.

##### 4.1 Split Perturbations

The process to split perturbations is quite simple. For each grid it designs, a new perturbation is created using the components associated with that grid while the other components are ignored.

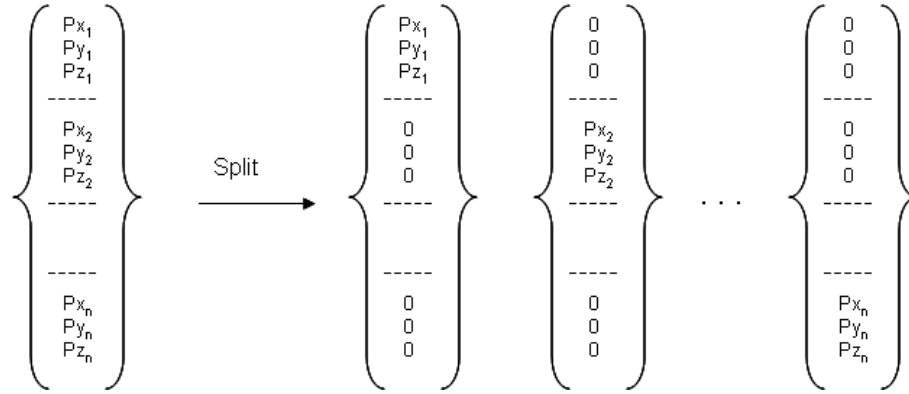


Figure 3. Perturbation Split

##### 4.1.1 Split Perturbation Example

The following example shows a perturbation vector that is split into multiple ones. A flat plate is used. Fig. 4(a) shows the initial perturbation vector that is used to simultaneously design 25 grids. Fig. 4(b-d) shows 3 of the 25 perturbation vectors generated. In these figures, the color bars represent the magnitude of the perturbations. Red indicates that the grid has moved to its maximum location and blue indicates that the grids have not moved.

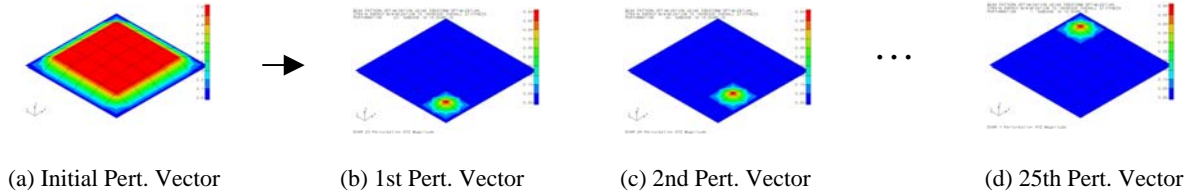


Figure 4. Split Perturbation

##### 4.2 Grid fraction Constraints

Grid fraction constraints are used to limit the number of grids that can move in the designable space. These constraints typically create more economical and clean designs. There are several optional expressions to implement this type of constraint. Here we will only show one of them:

$$gfr(x_1, x_2, \dots, x_n) = \frac{\sum_{i=1}^n |x_i|}{UB_{gfr}} - UB_{gfr} \leq 0.0 \quad (2)$$

where,  $gfr$  is the grid fraction constraint,  $x^i$  is the  $i$ th design variable,  $n$  is the number of design variables in the freeform region and  $UB_{gfr}$  is the upper bound value of the constraint.

Typical values for the grid fraction upper bounds are 0.2, 0.35 and 0.5. These values are for limiting the number of grids that move to 20%, 35% and 50%, respectively. The third example, in section 8 of this paper, shows the effect of using different grid fraction constraints.

##### 4.3 Distortion Constraints

The main drawback of shape optimization and any other special case of it such as topography or freeform optimization is that the elements of the mesh can get distorted. Distorted elements reduce the accuracy of the finite element results and produce answers that are typically stiffer than the correct answers. Moreover, distorted elements might yield to negative or zero jacobians of the stiffness matrices producing either unsolvable system of equations or incorrect answers. For these

reasons, it is important to prevent this problem. The traditional way to solve this problem is by either using mesh smoothing where the grids of the mesh are internally moved to reduce distortion or by simply reducing the scope of the design variables. The first method is very useful, while the second method is inconvenient and limiting because it reduces the chance to improve the design. In the Genesis program, we added the ability to automatically create a distortion constraint so the problem is attacked directly. Each type of measurable distortion parameters, such as the jacobian of the stiffness matrix, aspect ratio, interior angle and warping, are considered. The type of equation used is the following:

$$gd_l(x_1, x_2, \dots, x_n) = \frac{Distortion_l(x_1, x_2, \dots, x_n) - UB_{gd_l}}{UB_{gd_l}} \leq 0.0 \quad (3)$$

and/or

$$gd_l(x_1, x_2, \dots, x_n) = \frac{LB_{gd_l} - Distortion_l(x_1, x_2, \dots, x_n)}{LB_{gd_l}} \leq 0.0 \quad (4)$$

where,  $gd_l$  is a grid distortion constraint associated to  $Distortion_l$ ,  $x_i$  is the  $i$ th design variable,  $n$  is the number of design variables in the freeform region,  $UB_{gd_l}$  is the upper bound value of the constraint and  $LB_{gd_l}$  is the lower bound value of the constraint.

The values for the distortion bounds are element dependent and distortion type dependent. These constraints are generated automatically. These constraints are also used with shape and topography optimization.

#### 4.4 Manufacturing Constraints

If there are manufacturing constraints, such as symmetry constraints, the split process is changed to ensure that valid perturbations are created. Examples of manufacturing constraints are: planar symmetries, cycle symmetries and extrusion. Examples in sections 6 to 9 show the use of manufacturing constraints.

#### 4.5 The Freeform Optimization Problem

The freeform optimization problem with distortion and grid fraction constraints can be stated as:

$$\begin{aligned} & \underset{x_i}{Min} F(x_1, x_2, \dots, x_n) \\ & \text{such that :} \\ & g_j(x_1, x_2, \dots, x_n) \leq 0; \quad j = 1, m \\ & gfr_k(x_1, x_2, \dots, x_n) \leq 0; \quad k = 1, ngf \\ & gd_l(x_1, x_2, \dots, x_n) \leq 0; \quad l = 1, nd \\ & xl_i \leq x_i \leq xu_i; \quad i = 1, n \end{aligned} \quad (5)$$

where  $F$  is the objective function,  $g_j$  are the standard constraints,  $gfr_k$  are the grid fraction constraints,  $gd_l$  are the distortion constraints,  $x_i$  is the  $i$ th design variable and  $xl_i$  and  $xu_i$  are the lower and upper side constraints.

This optimization problem is solved using approximation concepts described in section 2.

### 5. Advantages and Disadvantages of Freeform Optimization with respect to Other Optimization Types

#### 5.1 Freeform vs. Shape Optimization

The main advantage of freeform over shape optimization is that it can produce more variability than shape optimization. The main disadvantage of freeform over shape optimization is that it has more chances than shape optimization to distort the finite element meshes.

#### 5.2 Freeform vs. Topography Optimization

The main advantage of freeform over topography optimization is that it works with any type of elements. Topography is mostly used with shells and occasionally with solids. Another advantage of freeform over topography is that freeform can generate beads with variable heights. See section 9 of this paper for an example which illustrates this latter mentioned advantage.

#### 5.3 Freeform vs. Topology Optimization

The main advantage of freeform over topology optimization is that it can add extra material outside the borders. Topology optimization can only take material away. The main advantage of topology over freeform is that topology can produce interior cavities and that it can not distort meshes.

## 6. Example. Rib Pattern Optimization of a Solid Structure

### 6.1 Description of the Problem

This example demonstrates the use of freeform optimization to find the optimal location of rib patterns in a solid structure. The overall dimensions of the structure are 40 mm x 18 mm x 1.5 mm. The structure is assembled with 8640 hexahedral elements and 11988 grids. The Young's modulus is 207,000 MPa and the Poisson's ratio is 0.3. There are three torsion load cases. In the first load case, one end is fixed on its two corners while the other end is subject to a pair of loads that produce a twist. In the second load case, the loads and boundary conditions are reversed from the first load case. In the third load case, the four corners of the structure are constrained while the center of the structure is loaded in torsion. The three load cases can be seen in Fig. 5.

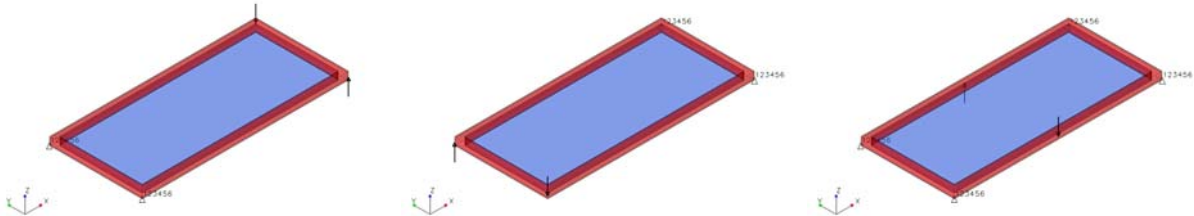


Figure 5. Load cases

A perturbation vector that can move the top and bottom surface grids is applied as shown in Fig. 6. The colors in this figure show the magnitude of the perturbation. The red color represents the maximum movement which is 1.2 mm, the blue color represents no movement. In Fig. 6 grids in red are designable while the grids in blue are not designable. Fig. 6 can show the components of the perturbation vector in the top. The components in the bottom are similar to the ones in the top but instead of pointing upward, they point downward making the perturbation vector symmetric with respect to the midsize x-y plane that passes through the center of the structure.

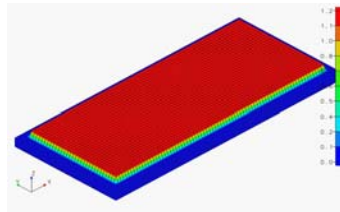


Figure 6. Perturbation Vector

The objective function of the problem is to minimize the sum of the normalized strain energy of the three torsion load cases. A grid fraction constraint of 0.45 is used to reduce the number of grids that move. In addition, manufacturing constraints are used to obtain triple symmetry about the center of the structure.

### 6.2 Results

Fig. 7 shows four views of the freeform optimization results. The final design is triple symmetric, as imposed, and the 45% grid fraction seems to be satisfied. The output file of the program, not shown here, confirms that.

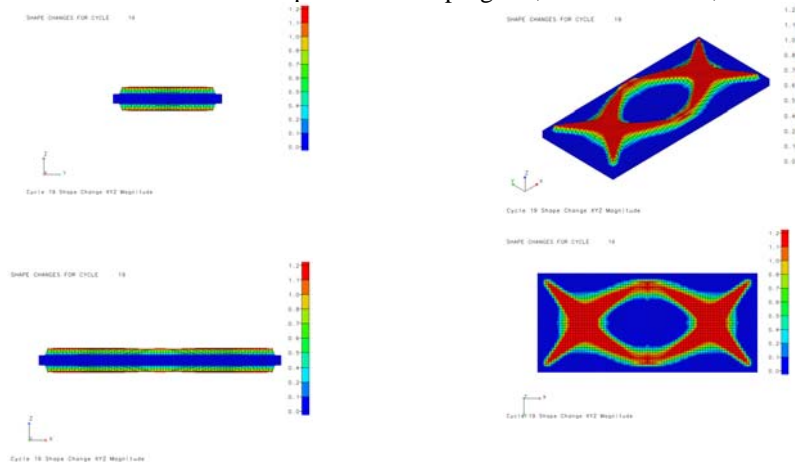


Figure 7. Freeform Results

Using the provided perturbation vector, shown in Fig. 6, freeform optimization automatically generated 608 perturbation vectors and 608 corresponding design variables. These 608 perturbation vectors controlled the design of 4650 surface grids. A plot of the objective function versus the design cycle number can be seen in Fig. 8(a). The objective function was reduced from 3.0 to 0.64. In other words, the final structure is nearly 5 times stiffer than the original one. Fig. 8(b) shows that the normalized mass changed approximately 60%. In this case, the optimization process took 19 design cycles to converge. At the end of the optimization there were no violated constraints.

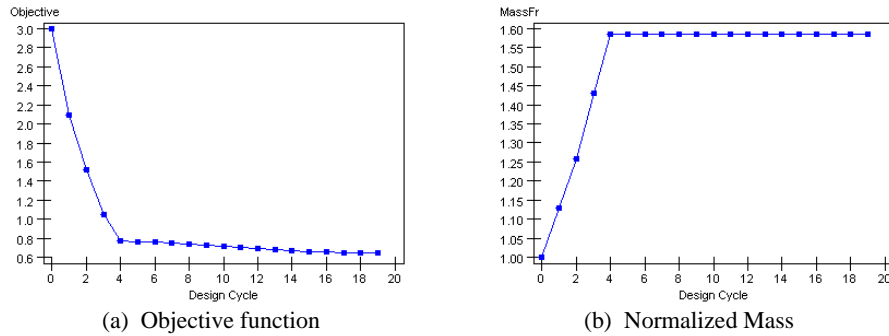


Figure 8. Optimization History Results

### 6.3 Validation of the Problem

To validate the freeform optimization results of this problem, a topology optimization run is performed on an equivalent problem. Since topology optimization can not add material, two layers of elements are added to the top and bottom of the structure. The topology model is assembled with 18,368 hexahedral elements and 22,228 grids. The topology model uses the same objective function and manufacturing constraints. In addition, a mass fraction constraint of 0.45 is used. The topology designable area was limited to the added material area.

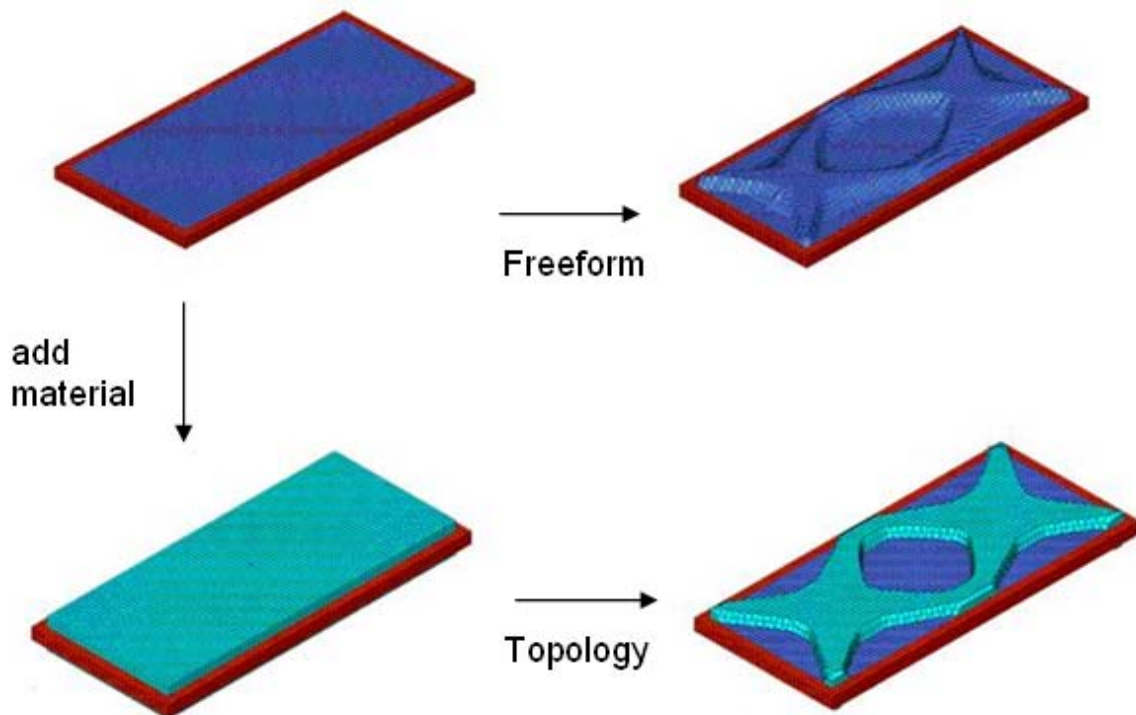


Figure 9. Freeform Optimization vs. Topology Optimization Designs

### 6.4 Comparison of Freeform and Topology Optimization Results

Freeform and topology optimization initial and final designs are shown in Fig. 9. From this figure, it can be seen that both freeform and topology optimization results suggest similar rib patterns. This topology optimization result validates the freeform results.

## 7. Example. Bead Pattern Optimization Using Different Types of Manufacturing Constraints

### 7.1 Description of the Problem

This example demonstrates the use of freeform optimization for finding the optimal location of bead patterns using different types of manufacturing constraints. The example uses a plate structure. The overall dimensions of the plate are 40 mm by 18 mm. The thickness of the plate is 1 mm. The structure is assembled with 720 quadrilateral elements and 779 grids. The Young's modulus is 207,000 MPa and the Poisson's ratio is 0.3. The plate has two corners constrained in one end and it is subjected to torsion loads in the other end, as shown in Fig. 10(a), below. A simple perturbation vector is created using a uniform pattern as shown in Fig. 10(b). The objective function of the problem is to minimize the strain energy to maximize the stiffness of the structure. There are three alternative designs and they are described below.

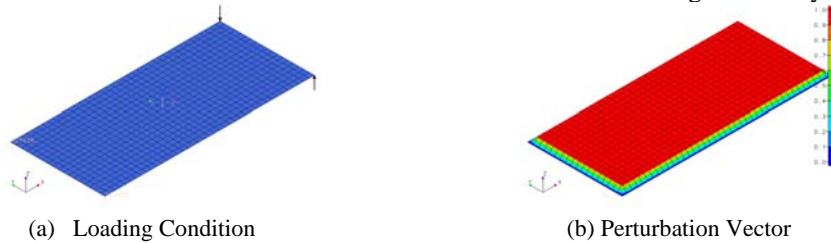


Figure 10. Inputs to the problem

### 7.2 Alternative Design 1. Symmetrical Design

In this case a manufacturing constraint is used to enforce symmetry with respect to the x-z plane that passes through the center of the plate. A bead fraction constraint of 0.33 is used to limit the amount that the grids can move. In addition, the grids are restricted to move only upward.

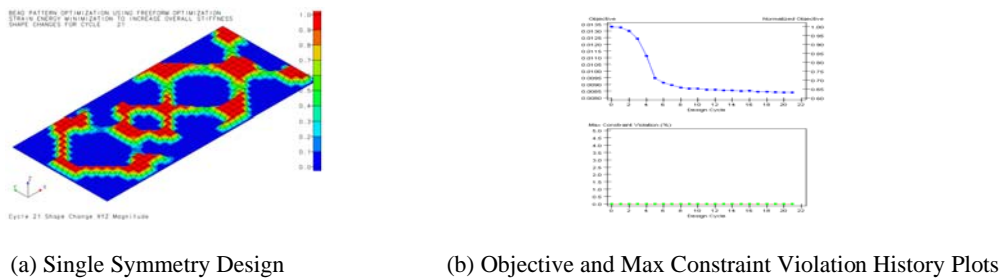


Figure 11. Freeform Results for Single Symmetry Design

For this alternative, freeform optimization automatically generated 351 perturbation vectors and 351 corresponding design variables. Fig. 11(a) above, shows the final design. Fig. 11(b) shows that it took 21 design cycles to converge and that there are no constraint violations in the last design cycle. The strain energy decreased approximately 35 percent.

### 7.3 Alternative Design 2. Double Symmetry Design

In this case, two manufacturing constraints are used to enforce double symmetry with respect to the center of the plate. A bead fraction constraint of 0.33 is used to limit the amount that the grids can move. In addition, the grids are restricted to move only upward.

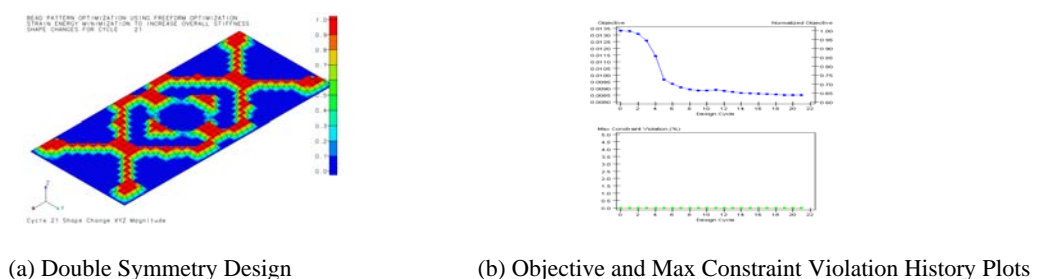


Figure 12. Freeform Results for Double Symmetry Design

For this alternative, freeform optimization automatically generated 180 perturbation vectors and 180 corresponding design variables. Fig. 12(a), above, shows the final design. Fig. 12(b) shows that it took 21 design cycles to converge and that there are no constraint violations in any design cycle. The strain energy decreased approximately 35 percent.

#### 7.4 Alternative Design 3. Uniform Design

In this case, two manufacturing constraints are used. One manufacturing constraint is used to enforce symmetry with respect to the x-z plane and another to generate a uniform profile throughout the x direction. A bead fraction constraint, equal to 0.50, is used to limit the amount that the grids can move. Here the equality constraint is used to induce the optimizer to make the answers more sharp.

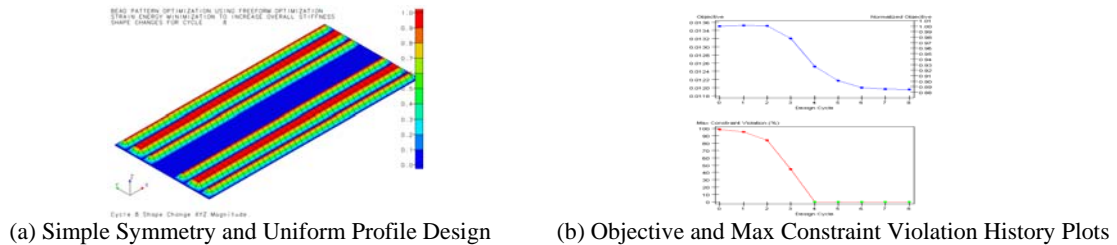


Figure 13. Freeform Results for Simple Symmetry and Uniform Profile Design

For this alternative, freeform optimization automatically generated 8 perturbation vectors and 8 corresponding design variables. Fig. 13(a), above, shows the final design. Fig. 13(b) shows that it took 21 design cycles to converge and that there are no constraint violations in the last design cycle. The strain energy decreased approximately 11 percent.

#### 7.5 Comparing the Alternative Designs

Alternative designs 1 and 2 are similar in a way that both produced beads which are oriented in 45 degrees with respect to the main direction of the plate. These two results are reasonable as when there are torsion loads experience has shown, that they can be carried to the supports more effectively with members distributed in 45 degrees. In design 3, where the beads are forced to be along the main direction (x), the results are consistent with the manufacturing requirement. The first two alternative designs are more efficient to carry torsion loads than the last design, they can improve the objective function more (35% versus 11% improvement).

### 8. Example. Bead Pattern Optimization of Curve Structure Using Different Types of Grid Fraction Constraints

#### 8.1 Description of the Problem

This example demonstrates the use of freeform with different bead fraction constraints. The example uses a curve shell structure assembled with 3200 quadrilateral elements and 3381 grids. The overall projected-dimensions of the structure are 160 mm by 50 mm by 20 mm. The curvature of the structure is obtained by making the height of the structure a quadratic function of x, where x is measured along its length. The thickness of the plate is 1 mm. The Young's modulus is 207,000 MPa and the Poisson's ratio is 0.3. The shell has two corners constrained in one of the short sides and it is subjected to torsion loads applied to the opposite short side. The objective function of the problem is to minimize the strain energy. Three alternative grid fraction constraints are imposed: 0.15, 0.20 and 0.25. These grid fraction constraints are to allow the grids to move up to 15%, 20% and 25% respectively.

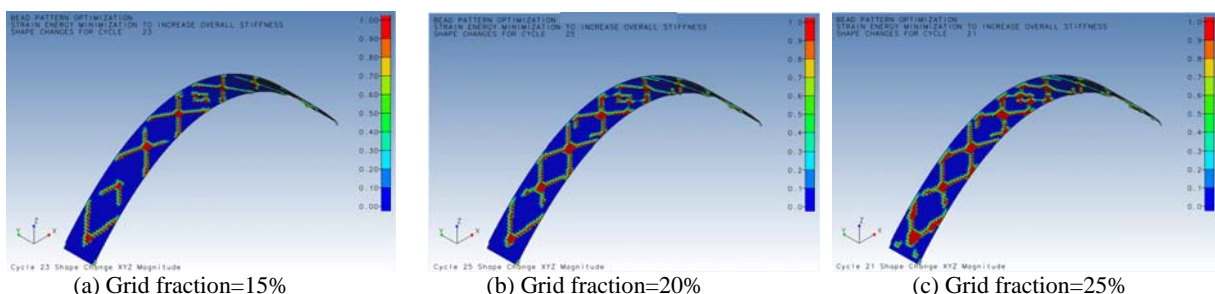


Figure 14. Freeform Results Using Different Grid Fraction Constraint Values

#### 8.2 Comparing the Alternative Designs

Figures 14 (a-c) show that when the grid fraction constraints are increased, the number of beads increases. These results are expected as the role of bead fraction is precisely to do that.



## 9. Example. Bead Pattern Optimization Using Non-Uniform Maximum Height

### 9.1 Description of the Problem

This example demonstrates the use of freeform to obtain bead patterns that can have non-uniform heights. The objective function of the problem is minimizing the strain energy. There is a grid fraction constraint of 0.3. There are manufacturing constraints: the first one is to enforce symmetry with respect to the plane x-z that passes through the center of the plate and which divides it in two throughout the length and the second is to obtain longitudinal beads. To obtain bead patterns that are not uniform in height, the only thing to do is generate a perturbation vector that represents an upper bound for the design. The initial design is shown in Fig. 15(a). The provided non-uniform perturbation vector is shown in Fig. 15(b).

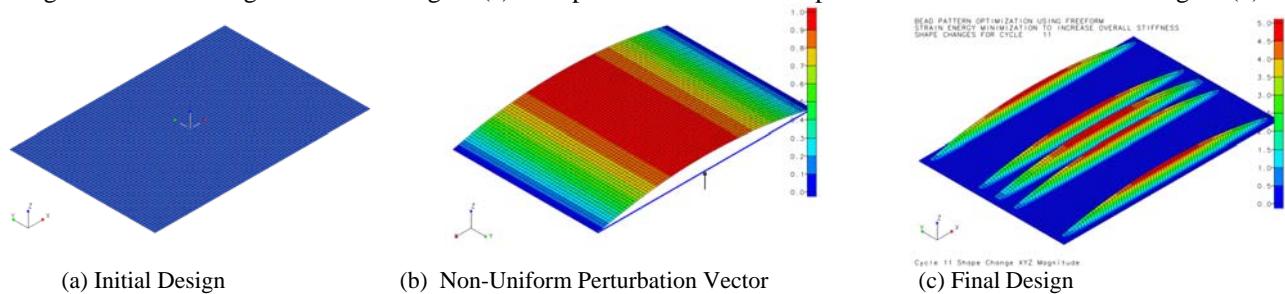


Figure 15. Optimal Location of Bead Patterns Using Freeform Optimization

### 9.2 Results

The final design shown in Fig. 15(c) has mirror symmetry, longitudinal symmetry and bead patterns with variable height, as required. It is interesting to mention here that this result can not be obtained with topography optimization.

## 10. Summary and Conclusions

This paper has described freeform optimization which is a new capability to optimize structures. This new capability can be defined as a grid-by-grid shape optimization capability. Three special constraints that improve the results were discussed: distortion constraints, grid fraction constraints, and manufacturing constraints. This capability can be used with any type of structure but is particularly useful to find ribs in solid structures and uniform or non-uniform bead patterns in shell structures.

## References

1. Leiva, J. P., Topometry Optimization: A New Capability to Perform Element by Element Sizing Optimization of Structures, presented at the 10th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Albany, NY, August 30 - September 1, 2004.
2. Leiva, J.P., "Methods for Generation Perturbation Vectors for Topography Optimization of Structures" 5th World Congress of Structural and Multidisciplinary Analysis and Optimization, Lido di Jesolo, Italy, May 19-23, 2003.
3. GENESIS User's Manual, Version 11.0, VR&D, Colorado Springs, CO, November, 2009.
4. Leiva, J.P., and Watson, B.C., "Automatic Generation of Basis Vectors for Shape Optimization in the Genesis Program," 7th AIAA/USAF/NASA/ ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, MO, Sep 2-4, 1998, pp. 1115-1122.
5. Design Studio for Genesis, Step-by-Step Example Manual, Version 11.0. Vanderplaats Research and Development, Inc., Colorado Springs, CO, October 2009.
6. Schmit, L. A., and Farshi, B., "Some Approximation Concepts for Structural Synthesis," AIAA J., Vol. 12(5), 1974, pp 692-699.
7. Schmit, L. A., and Miura, H., "Approximation Concepts for Efficient Structural Synthesis," NASA CR-2552, March 1976.
8. Vanderplaats, G.N., and Salajegheh, E., "New Approximation Method for Stress Constraints in Structural Synthesis," AIAA J., Vol. 27, No. 3, 1989, pp. 352-358.
9. Canfield, R. A., "High Quality Approximations of Eigenvalues in Structural Optimization of Trusses," AIAA J., Vol 28, No. 6, 1990, pp. 1116-1122.
10. Vanderplaats, G. N., Numerical Optimization Techniques for Engineering Design; with Applications, 3rd Ed., Vanderplaats Research & Development, Inc., 1999.
11. DOT, Design Optimization Tools User's Manual, Version 5.0. Vanderplaats Research and Development, Inc., Colorado Springs, CO, January 1999.
12. Vanderplaats, G., "Very Large Scale Optimization", presented at the 8th AIAA/USAF/NASA/ISSMO Symposium at Multidisciplinary Analysis and Optimization, Long Beach, CA September 6-8, 2000.