Free-Form Grid Shell Design Based On Genetic Algorithms

ABSTRACT

In the 21st century, as free-form design grows in popularity, grid shells are becoming a universal structural solution, enabling the conflation of structure and skin (façade) into one single element (Kolarevic 2003). This paper presents some of the results of a comprehensive research project focused on the automated design and optimization of grid structures over some predefined free form shape, with the goal of generating a stable and statically efficient structure. It shows that by combining design and FEM software in an iterative, Genetic Algorithms-based optimization process, stress and deformation in grid shell structures can be significantly reduced, material can be saved and stability enhanced.

Milos Dimcic  
Stuttgart University

Jan Knippers  
Stuttgart University
1 Introduction

At the end of the 20th century we witnessed the appearance of the first steel free-form grid shell structures entirely composed of unique structural members, since there was no longer any substantial difference in cost between producing 1000 unique objects and 1000 identical ones (Kolarevic 2003). In the 21st century the field of free-form grid shell structural design is being developed further, but structural design and optimization techniques are still mostly based on the trial-and-error approach. In simpler terms, we developed a variety of techniques that enable us to generate optically acceptable triangular, quadrangular or hexagonal grids over a given free-form surface, but when their statical efficiency is brought to attention there are no ready answers about how to optimize the grid. This paper shows how by changing the member disposition, i.e., by performing geometrical and topological optimization of the grid shell, substantial differences in statical performance can be achieved. In order to not limit the creativity of architects, the idea was to generate the best structural solution over some already defined shape. Instead of form-finding we are trying to find the best geometry and topology of a grid shell, while keeping it on the specific surface during the process. The proposed method of structural optimization is constructed as a C++ based plug-in for Rhinoceros 3D, one of the main NURBS (Non Uniform Rational B-Splines) geometry based modeling tools used by architects for free-form design today. The algorithm communicates iteratively with FEM software for static analysis. In this case Oasys GSA commercial FEM software is used.

2 Grid Formation

Before the optimization algorithm explanation, the method of automatic grid generation over a given free-form NURBS surface has to be addressed. This is important in order to understand how different grid shell solutions are generated in the process of finding the most efficient one. For this purpose, and within the presented research, the decision was made to use Voronoi Diagrams (De Berg et al. 1997), for two main reasons. First, NURBS surfaces are mathematically represented over two parameters (uv) and algorithms for Voronoi diagram generation in 2D (in plane) can be therefore mapped onto the surface, using a direct xy-uv transformation. Second, depending on the disposition of Voronoi points, a large number of different, natural looking structures can be generated, but also structures with a regular grid pattern (like triangular, quadrangular and hexagonal). Therefore, Voronoi points generated over a given NURBS surface are basic variables. As depicted in Figure 1, we take the surface, generate a Voronoi diagram over it and what we do next is relax the Voronoi structure. For the process of relaxation the Force Density Method (Gründig et al. 2000) is expanded to work for any kind of grid, and additionally to always keep the grid on the surface, while relaxing it. By relaxing a Voronoi structure we get foam-like grid that we called Voronax (Voronoi + Relax). The Voronax grid has polygons (cells) with much more similar corner angles and edge lengths, which are, from a structural point of view, more acceptable for the grid shell design. The advantage of this complexity is that Voronax grids can easily change their density, while being optically smooth and structurally acceptable. They keep the topology of the Voronoi diagram which means that on average their polygons have ~ 6 edges (Sack 1999; Urrutia 1999). We can use that to see what distribution of density (distribution of structural members) is statically favorable.

3 Basic Plug-in Structure

The goal of this research is to make a universal method for grid shell optimization; one that is adaptable, easily expandable and with a large number of variables, (i.e., with an easy definition of boundaries and settings within which we want our solution to be generated). Therefore a plug-in was developed so that the user can:
1) Choose the surface over which the grid will be generated
2) Choose the basic pattern of the grid (e.g. Delaunay triangulation (De Berg et al. 1997), quadrangular, Voronoi, Voronax)
3) Set a support combination (e.g. all four edges, two edges, fully restrained, movable)
4) Set a load combination (any load combination definable in FEM software)
5) Set material properties
6) Set cross-section of the structural members
7) Define the fitness function (e.g. minimize Von Mises stress, minimize deformation, maximize load buckling factor)
8) Define one or more penalty functions (e.g. limit the length of a member, limit the size of a polygon, limit the stress generated in one member)
9) Set GAs parameters (e.g. crossover and mutation probability, number of individuals, number of generations) Each one of these settings (Figure 2) can be easily expanded and redefined. When they are chosen, the optimization process begins and the algorithm converges toward the best solution for that [combination of input settings, whatever they are].

4 Genetic Algorithms

Genetic Algorithms (GAs) are chosen as a suitable method for multi-objective and highly non-linear optimization. It is a stochastic method, based on the principle of evolution, within which a random population of individuals is generated (grid shells in our case) at the beginning. The best individuals, according to their fitness, are then chosen for reproduction and with specific crossing techniques, solutions are combined to bring new offspring and in that way form a new generation. The crossing methods ensure the heritage of good genes, thus enabling the whole process to converge toward the best fitness solution. Specific mutation algorithms enable random alteration of individuals in order to introduce diversity and ensure a better exploration
of the search space, thus avoiding convergence to local optima. This loop (Figure 3) then continues until the satisfactory solution is found. In our case, we are searching for a grid shell structure with minimum material usage (minimum weight) and minimum potential energy of the system. Grid shells can be evaluated optically or statically, according to the defined fitness function, and in this paper the focus is on the statical optimization. More on the basics of the Genetic Algorithms application can be found in Genetic Algorithms in Search, Optimization and Machine Learning (Goldberg 1989).

4.1 BASIC LOOP

Genetic Algorithms work with a chromosome representation. In this research the chromosome is formed as a string of real-valued numbers which are later on transformed into the $uv$ coordinates on the surface. This is done with a specific set of decoding functions. The $uv$ coordinates are used to generate points from which a Voronoi diagram (over a given surface) is calculated and eventually relaxed, resulting in a Voronax grid structure. Each grid shell in the algorithm goes through an eleven step process depicted in Figure 4. First, the basic GAs operations (selection, crossing, mutation) are performed, followed by the decoding part (or generation) where the chromosome is transformed into a grid shell and prepared for FEM static analysis. Step 8 refers to an automatic call of the FEM software where the static analysis of the generated grid shell is performed. When the needed results are obtained (e.g., forces, moments, deformations, etc.) the evaluation according to the chosen fitness function is carried out, and the solution is penalized if it violates any of the specified constraints. The fitness value and the violation of constraints are then combined and scaled into one final fitness value of the generated individual solution. In a usual optimization there are 50 grid shells in a generation, and the process lasts for 400-700 generations, thus sometimes generating more than 30,000 solutions. All the solutions are kept in specific text files that enable their recreation, i.e., extraction and drawing of any of the generated grid shells in the process.

5 Optimization

In order to illustrate the optimization process, and what its contribution is, a surface shown in Figure 5 is chosen. It is a free-form vertical wall, the edges of which are restrained, i.e., the structural joints of the generated solutions on the edges are restrained from movement or rotation in all directions. In Figure 5 we also see a basic cross-section used for the optimization, the circular hollow section: CHS 193x5.0. The idea is to perform a geometrical and topological optimization of the grid, and therefore all generated members have the same section. In that way we can look for the minimal stress or minimal displacement solution by changing the geometry and keeping the mass of the structure relatively the same. The load applied is the self-weight of the structural members and a horizontal surface load. The horizontal load is applied by calculating the surface of each cell (structural polygon), and distributing it to the structural joints (Figure 5). Within the research, experiments were done with properly oriented rectangular cross-sections and with proper wind load (normal to the surface at all points). An optimization with these settings however introduces a different set of problems which are not the focus of this paper, and that is why, for the presented optimization, the settings were simplified using a circular section and horizontal load. This however has no effect on the efficiency of the optimization process, since it works for any kind of input parameter combination.

The most important part of the GAs optimization is the fitness function. In this case the goal is to minimize Von Mises stress ($\sigma_v$) in the structure. For each structural member in the grid shell the simplified version of Von Mises stress (Equations 1-4) is calculated at both of its ends (denoted

Figure 5. Surface, cross-section and load
as 0 and 1). Those values are summed up for all \( n \) structural members resulting in a fitness value \( F(x) \) for the entire structure, which we are trying to minimize (Equation 5).

\[
\text{Eq. 1} \quad \sigma_s = \sqrt{\sigma_{x}^2 + 3\tau_{x}^2 + 3\tau_{y}^2} \\
\text{Eq. 2} \quad \sigma_s = \frac{F_s}{A_s} = \frac{M_z}{W_z} + \frac{M_y}{W_y} \\
\text{Eq. 3} \quad \tau_{x} = \frac{F_t}{A_t} \\
\text{Eq. 4} \quad \tau_{y} = \frac{F_t}{A_t} \\
\text{Eq. 5} \quad F(x) = \sum_{i=1}^{n} (\sigma_{x,i} + \sigma_{y,i})
\]

Here we also introduce another fitness function developed within the research, which will be used only for comparison purposes. Namely, for each joint in the structure its displacement (movement) is calculated (di) as a vector in space, derived from the movements in all three \((x,y,z)\) directions (Equation 6). The magnitude of all joint movements is then summed up, resulting in a total displacement of the structure (Equation 7).

\[
\text{Eq. 6} \quad d_i = \sqrt{x_i^2 + y_i^2 + z_i^2} \\
\text{Eq. 7} \quad F(x) = \sum_{i=1}^{n} d_i
\]

5.1 VORONAX OPTIMIZATION

The Voronax pattern optimization is performed with a 150 point chromosome. That means that for each individual solution, 150 points are generated over a surface, turned into a Voronoi diagram, which is then relaxed resulting in a Voronax grid structure. In Figure 6 there are two graphs showing the convergence of the optimization process after 550 generations (27,500 generated individual grid shell solutions). The graph on the top shows the progress of the average fitness value in each generation (calculated from 50 individuals). The graph below shows fitness values of the best individual solution (grid shell) in each generation. It can be seen how both graphs show a constant descent of the total Von Mises stress generated in the structure and a steady convergence.

In the middle column, depicted from the front view, there is:

1. The worst generated solution, created randomly in one of the first generations, having 113 GPa as the total amount of Von Mises stress and 13.4m of total joint displacement.

2. For comparison, a hexagonal structure is used, representing basically a uniform version of the Voronax grid. The reason for this is that Voronax keeps the topology of the Voronoi structure after relaxation, which means that on average its polygons have ~ 6 edges and joints have a 3-member connection (as in a hexagonal grid). This uniformly distributed grid only shows a slightly better performance (101 GPa and 7.58m) than the worst generated solution.

3. The best generated solution from one of the latest generations has the smallest amount of Von Mises stress generated in its members (38 GPa), i.e., three times smaller than the worst generated solution and a 6 times smaller amount of displacement (2.22m). In Figure 6, on the right-hand side, there is a colour analysis of this Voronax grid solution, showing the distribution of the grid density (from blue=sparse to red=dense).

There is a number of different ways of how this information can be used in grid shell design. Following the advice of the GAs algorithm we can use different techniques, from controlled relaxation to a combination of different patterns, to achieve a statically efficient design. The following is an examination of such a design.

5.2 INTERPRETATION

We can generate a uniform quadrangular structure over our free-form wall as shown on the left-hand side in Figure 7. Then we can try to interpret the intention of the GAs optimization process. It can be seen that the best structural solution offered has an enlarged grid density around the convex parts.
[red area in two representations in the middle of Figure 7] thus stiffening them up, and stretching the cells over the diagonal between the two convex parts (yellow area). Using this information we can try to generate a quadrangular structure with a similar number of joints and members, as depicted on the right-hand side of the figure. By doing so, we get a quadrangular structure with 13% less generated stress and a 25% smaller amount of displacement. By combining different patterns (triangular, quadrangular, hexagonal) we can develop different solutions, knowing the distribution of grid density (hence stiffness) that produces optimal results according to the desired criteria.

6 Conclusion

This paper presents an automated method of grid shell optimization that offers optimal structural solutions over some given free-form surface. The focus is on the fact that no approximation or pure trial and error method has to be involved in the structural design process if we use the proposed optimization method. The advantage of the Voronax structure is that it can be easily interpreted most of the time. For example, in Figure 8, there are results of the optimization done over two flat vertical surfaces, with the same load combination applied as in the examples above (self-weight of the structural members + horizontal load). In the example on the left, the joints are restrained on four corners of the structure, and in the middle of the surface edges on the structure depicted on the right (restrained areas are marked red). For each option the best solution obtained in an optimization process can be seen, and next to it a look through the last generation is depicted. Namely, if we take all 50 solutions of one generation and line them up one behind the other, we can get a comprehensive picture of the intention of the optimization process. It can be seen how the center part in both cases has larger cells, stabilized with the O-shaped formation of denser cells in the case on the left and the X-shape formation in the case on the right.

These experiments are a part of the comprehensive research done with different shapes, fitness functions, penalty functions, support and load combinations and different patterns. Optimizations are done not only as single-objective but also as multi-objective ones, showing that, depending on the free-form shape and grid pattern, we can generate grid shells that have up to 6 times less Von Mises stress and up to 10 times less displacement when compared to a regular (uniform) structure, generated with the same number of structural members and over the same given surface.

References


