PriMera Scientific Engineering
Volume 4 Issue 5 May 2024
ISSN: 2834-2550

Optimized Form-finding in Grid Shell Structures, Using Finite Element in Genetic Algorithms

Type: Literature Article
Received: March 14, 2024
Published: April 26, 2024

Citation:
Faezeh Choobkar., et al. "Optimized Form-finding in Grid Shell Structures, Using Finite Element in Genetic Algorithms". PriMera Scientific Engineering 4.5 (2024): 20-33.

Faezeh Choobkar[1]*, Morteza Rahbar[1], Alireza Gogani[2], Mohsen Faizi[1] and Abbas Tarkashvand[1]

[1]Faculty of Architecture and Environmental Design, Iran University of Science and Technology, Tehran, Iran

[2]Faculty of Civil Engineering, Bauhaus-Universität Weimar, Weimar, Germany

***Corresponding Author:** Faezeh Choobkar, Faculty of Architecture and Environmental Design, Iran University of Science and Technology, Tehran, Iran.

Abstract

Shells are optimized structures especially appropriate for covering great spans with minimum consumption of material. Besides, due to their free forms, they are aesthetically attractive. However, the process of form-finding in these structures, inherently relying on their structural performance, could be so intricate, since there may be some shells produced that are beautiful but unbuildable. Hence in this research, regarding some architectural parameters such the location of different functions, and the structural characteristics, we aim to develop a method for resulting suitable forms in shell structures. The given shell in this research is a 3D reticulated shell. Here, we will use a performance-based design method to introduce a new three-step method for producing optimized forms in regard to their performance and structure. The three steps are including manufacturing the primary form, evaluating its structure applying the finite element, and optimizing structural details with the help of Genetic Algorithms. Inputting special characteristics of each project, the algorithms presented in the Grasshopper plug-in could be generalized to other questions concerning form-finding in shells.

Keywords: shell structure; finite element method; structural optimization; genetic algorithm

Introduction

Advantages such as having free forms, the ability to cover great spans without using columns, and minimum consumption of materials, have made the shell structures very popular among designers (Paoli. 2007). These structures are 3D surfaces that could tolerate the loads with the benefit of their geometry (Adriaenssens et al. 2014). Their shape has a significant impact on their structural performance (Jiang. 2015). More the shell material sustains a load due to physical situations, the more the shell is stable (Hussain et al. 2020). Since in shells, there is no duality between architectural form and structural behavior, necessarily when it comes to shells, architects should prefer forms that have acceptable structural behavior. The architect would fail in this duty unless having some structural knowledge. If the form-finding process is done decorously for shells, then they could transfer the loads via axial stresses (Adriaenssens et al. 2014). This could be a huge challenge for architects during the rudimentary steps of design. In the past, the only tools that architects could have used during a design process were pen, paper, and rulers. The traditional method was based on additive logic in a way that architects gradually completed their designs with simple acts of adding or reducing some parts. However, still we have the problem of not taking into account the limitations such as loads that affect the building and make the structure deform or move. On the other hand, some of the form-finding methods such as Catenary, Soap Bubble, and Frozen Fabric that were developed respectively by Antonio Gaudi, Frei Otto, and Heinz Isler have had their own restrictions, and despite the relative control over the final form, the designer still, in these cases, is so confined by the restrictions of the natural form-finding methods. At first, this problem was not solved even with help of the computer, because the most common architectural software, the AutoCAD, did not (and still does not) key in the way forms are shaped. Since the beginning of the 1960s, there was a tendency grown among the avant-garde architects to consider these restrictions in their form-finding process, and as a result, they developed a system that enabled them to produce and modify their forms considering both architectural and structural standards. (Tedeschi.2014) These matters in the case of structural systems such as shells, in which the architectural form and the structure are the same and one, are more important than other systems. Therefore, in the current research, we have used a performance-based design method for form production. In this method, we could see the potentials of form production and evaluation as a consistent question (Oxman. 2008).

So, designing a shell, considering different aspects of architecture and structure, is an essential, unless complicated duty. Also, since a major part of construction cost spends on the structure, designing a materially optimized structure is of great importance. The advents in computer software for 3D modeling endowed the architects the ability to draw any imaginable form. But finding an optimized form in shells in a way that the form be both geometrically stable and responsive to architectural requirements, is a challenging problem. Today, computational methods through optimizing processes could help designers finding more appropriate responses. Many real-world decision problems involve simultaneous optimization of multiple objectives. Such problems are known as Multi-Objective Problems (MOPs) (Kaveh and Bakhshpoori. 2006) However, the usual optimization algorithms, ultimately, would result in a single response. In that way, these algorithms are not ideal for us, since they restrict the designer`s freedom and somehow reject his/her right to choose (Winslow et al. 2010).

That is why the current research promotes a back-and-forth method with three steps for form-finding based on structural performance. This method, considering architectural restrictions which are important for the designer, and using a multipurpose genetic algorithm, could provide the architect with many optimized formal alternatives and give him/her the right to choose between one of them. In this method, the architect during two steps of form-finding process, would have a freedom in taking his/her own actions; he not only imposes his preferred limitations at the first step, but also could choose the most aesthetically desirable and optimized alternative of the final forms, at the end of the process.

Indeed, here we aim to highlight the architect`s role in designing the structure and allow him/her, using advanced technology and designing appropriate forms, to analyze stresses and minimize the mass of elements. Besides, he/she, using the most optimized geometry, could find a financially optimized shell which is as well as buildable; so, in this way, the process of designing form and structure, which in these structures are deeply interrelated, could be analyzed and surveyed by the architect him/herself.

Literature Review

We have a long history of designing and constructing curvilinear structures in architecture. In the past, when it came to shell structures, there were few and limited ways for form-finding. In the traditional world, we built these structures with the help of physical methods, and then after some advances in our computational knowledge, we started using new methods based on these knowledges. Fortunately, the nowadays advances in computer sciences and their potentials for producing more problematical shapes, provided architects with new opportunities (Pillwein et al. 2020).

It worth mentioning the Church of Colònia Güell, designed by Gaudi, as one of the most momentous examples of shell structures. In this church, Gaudi realized the method of inversing Catenary for physical form-finding. In this way, with the help of Hooke`s law and using some cables and sandbags, he could have modeled the total weight of the church in real size. According to this law, when we inverse the catenaries which are under pure traction, they become curvilinear structures behaving as they are under pure compression (Jiang. 2015).

The idea of reticulated shell structures was first introduced by Shaunkhov in his design for the Pan- Russian exhibition in 1986 (Pillwein et al. 2020).

After Shankhov, it was Frei Otto who developed the concept of reticulated shells. During the late 1950s, Otto became interested in lightweight shells. Applying Hooke`s law, inversing catenaries in the way that Gaudi had done, and providing soap films, Otto was developing the form-finding process in his designs. Otto in 1975, designed Mannheim Multihalle (Liddell. 2015) which was supposed to be a provisional structure at first, but has rooted in its place until today. Many designers have used the catenary model and other physical methods to find the most efficient structural form which could perform in pure compression. Heinz Isler for more than 40 years has been promoting the use of physical models as the most appropriate method for 3d form-finding. Kilian and Ochsendorf (2005) aiming for this purpose, he has improved the use of frozen suspended fabrics.

It seems that there was an interruption in the history of designing shell structures with this method, till 2002 that Edward Cullinan, making some advances in structural details and construction, designed the reticulated structure of The Weald and Downland somewhere near Chichester, London. The accomplishment of the building project of Savil in 2006, designed by Glen Howells, improved the application of these methods (Tang. 2013).

During the subsequent years, the breakthroughs in computer sciences and parametric software, made the old way to be replaced with the new ones. High powered computers and engineering computer systems allow designers to routinely simulate complex physical phenomena (Fedorik et al. 2015). Since 2010, Navier Laboratory focusing on new materials and expanding the implication of mathematical methods has extended the study of reticulated shell structures. Navier lab had done some experiments on two primary cases based on their findings and attaining some experience, practiced them on the roof of their project for Solidays Festival (Du Peloux et al. 2016). In this structure, form-finding process was carried out with the help of computational methods (Du Peloux et al. 2013). In 2012, the provisional church of Créteil provided an opportunity for developing this new idea. Although the area occupied in this project was equivalent to the former one, it should have dealt new challenges. Despite the former projects, these two projects employed computational methods instead of physical ones (Du Peloux et al. 2016). Among all the contemporary architects who started designing shell structures using the computer, it worth mentioning Frank Gehry and Zaha Hadid.

Despite many advantages, there have been not so many of such structures built all around the world. The reason is that their form-finding process is too complicated. In this research, we aim to accelerate the form-finding process with the help of computational design and meanwhile to find some forms which respond to the structural and architectural restrictions, in their most optimized situation.

Methodology

The current study, employing the performance-based design method, conducting some analysis based on finite element method, and applying optimization genetic algorithms, is looking for a practical to find optimized shell forms. The main objective is to present an optimized way of form- finding based on architectural and structural parameters and to generate a form responsive to both fields. At the same time, because of the high number and complexities of these parameters, we have limited them in both fields. Regarding architecture, we have concentrated on the dimensions of different spaces based on their use, the place of these spaces, and suitable spatial proportion that is of major importance in shells and could hugely affect the final forms. Regarding structures, we have taken the cross-section of linear elements into account. The recommended method consists of three steps: generating the primary form, structural evaluation, using the finite element method, and optimizing that. These three steps are correlated together, and make a back-and-forth loop. (Fig. 1) This research, merging the aforementioned structural and architectural criteria, is trying to find a method for producing efficient forms with the help of genetic algorithms and some specified penalties in the procedure of form production.

Descriptions	***Symbols***
The dimensions of area	d (x), d (y)
The radiuses of spheres (variable 1)	R c
Random points	Pt (i)
The coordinates of the points (variable 2)	Xpt (i) , Ypt (i)
The radius of Structural cross-section (Variable 3)	R cs
Structural weight	W
The lateral load of wind	P
Structural displacement	M
Axial stresses	σ
Compression stresses	C
Tension stresses	T

Table 1: Nomenclature.

Form Generation

In the Performance-based design method, exploiting the digital design system in form-finding has three parts:

- To formula the primary geometric model in a way that could transform and generate forms based on inputs of the evaluation process.
- The evaluation processes could merge with geometric models and as a result, could generate the production and modification processes in the geometric model. These could be some single-objective evaluation (such as structural performance) or multi-objective evaluation consisting of multi-performance and optimizing factors.
- The Designer`s interactions with the system, will be maintained as a supervisor for different existing processes or as a designer of algorithm models for form production (Oxman. 2008).

First, to generate the primary geometric model, after surveying the site plan and physical program, we will clarify the number of spatial zones according to the function of space. After drawing the site plan in the environment of Rhino software with the help of the Grasshopper plug-in, in the specified area, four points (according to our four spatial zones) were randomly produced the coordinates of which, we considered as the first variable. Using the Cocoon plug-in, we will draw some spheres considering these points as their centers. To define the range in which the radius of our spheres could change, we investigated the maximum allowed occupancy rate, and minimum required area considering the height of space. And the result was: $20<R<30$. We see this as our second variable in our method. Since the proposed area for generating random points was the whole site plan, in some of the form-finding situations, if the resulted points were just on the boundaries, the spheres would have developed to out of the area. To solve this problem, we offset the boundaries of the area according to the size of the largest radius (Fig. 2).

After dividing the spheres into two different parts, using x-y surface and choosing upper semi- spheres, it was necessary to arrange the triangulation of resulted mesh, because in the step of structural evaluation each edge of these sections will be introduced to the Karamba3D plug-in as beams.

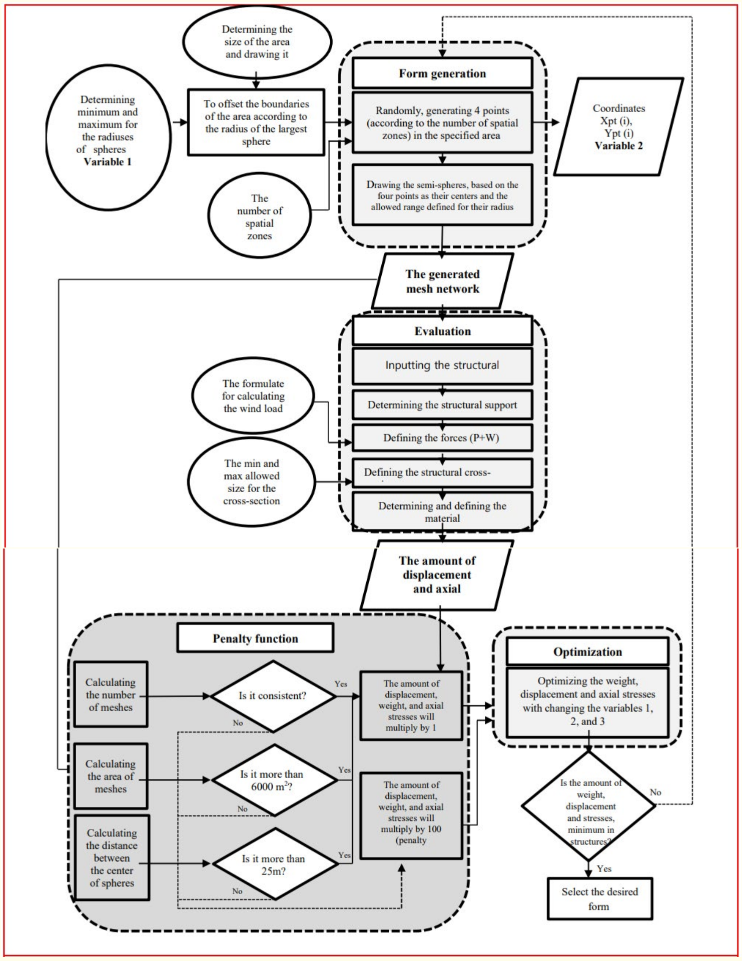


Figure 1: General stages of research.

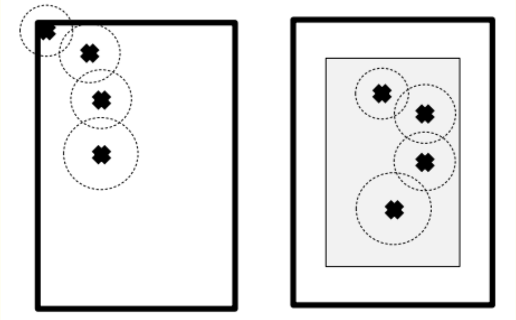


Figure 2: Determining the allowed area for form-finding.

Evaluation

In shells, the best structural performance will be realized when the structure is performing under axial stresses (Bletzinger and Ramm. 1993). Reticulated shell structures are kind of shells made up of a network instead of a solid surface (Malek. 2012). Indeed, beam-like structures that are correlated to each other, transfer the load to the structural support (Loukaides and Seffen. 2015).

The purpose of this research is to minimize the structural weight, the rate of stress in structural elements, and to limit the displacements. Using computational methods for analyzing complicated structures is so essential in the steps of design. Since the very distant past, we have been using the finite element method as a fundamental method in the computational analysis of shells (Bucalem and Bathe. 1997). The finite element method used in this research is consists of the linear static analysis method for calculating the structural load and the wind load. The structural analysis using this method have 8 general steps. First, by dividing the structure into smaller parts, the base element would be determined. The triangulated meshing network of the shell is made of pipe-like elements that connect to each other at the nodes. In the second step of the function, we will choose the appropriate shape for displacing the nodes.

$$U=\Sigma\ Ni\ di \quad (1)$$

Ni: shape functions.
Di: nodal displacement.
Determining the interpolation functions of displacement.

Step three consists of making a relationship between displacement of nodes and strain rate and a relationship between stress and strain rate. As the considered elements here are linear, for calculating strain rate (εx) and stress (σ), we use formulas 2 and 3.

$$x = d\,u/d\,x\ \varepsilon \quad (2)$$

How to calculate the strain rate.

$$\sigma=E\ \varepsilon x \quad (3)$$

How to calculate the stress.

Then to make a relationship between forces and node displacements, we will calculate the Stiffness Matrix for each of the elements.

$$\{F\} = [k]\ \{d\}$$
$$\begin{pmatrix} k_{11} & \cdots & k_{1n} \\ \vdots & \ddots & \vdots \\ k_{n1} & \cdots & k_{nn} \end{pmatrix} \begin{Bmatrix} d1 \\ \dots \\ dn \end{Bmatrix} = \begin{Bmatrix} f1 \\ \dots \\ fn \end{Bmatrix} \quad (4)$$

K: The Stiffness Matrix.
d: nodal displacement.
F: nodal forces.
The Stiffness Matrix for each of the elements.

In the next step, we will augment the Stiffness matrix of each of the elements, and with inputting structural support conditions, the stiffness matrix for the whole system would be calculated. On the other hand, when shaping stiffness matrix (by applying the conditions of adaptation) we should pay attention to elements that have mutual nodes.

$$\{F\}= [K]\ \{D\} \quad (5)$$

F: Nodal forces.

K: Total Stiffness Matrix.
D: Total nodal displacement.
The Stiffness Matrix for the whole system.

As normally "K" could not be invertible, we use the structural support condition and make the matrix invertible, and as a result, we find the linear algebra equation. Using different methods, we solve the algebraic equation and find the unknown displacements (D). After determining all the displacements of nodes, we will turn back to the elements, find the displacement of each element, and using the stiffness matrix of each of them, calculate the nodal forces and evaluate the stress and strain rate. In the last step, we interpret and analyze the results to design.

If we wanted to do the calculation for all the elements of the shell one by one, we should have spent a lot of time and accepted the higher risk of making mistakes. Also, we had 5000 formal instances, and we could not have done this procedure for all of the produced forms. Therefore, we used the Karamba3D plug-in, which is scripted based on the finite element, in the environment of the grasshopper. A finite element analysis in the Karamba3D plug-in has three steps consisting of pre-processing, processing, post-processing. Pre-processing is the process that transforms the geometric model into the structural model in which points will become nodes, lines will become beams, and the surfaces will become shells. Importantly, in this step, we should designate the structural support points. As here both orbital and rotational motion should be limited in all the directions (Tx=Ty=Tz=0, Rx=Ry=Rz=0), so we define the points of intersection between the shell and ground as fixed supports. Forces affecting this structure, are the weight force and lateral force. The structure is so lightweight, and our work has an aerodynamic form and creates a suction, so the wind load will be the lateral load that affects the structure the most. After investigating the guideline principles, we used Eq(6) to calculate the load of wind.

$$P = I_w q C_e C_g C_p \qquad (6)$$

Importance factor :Iw.
Velocity pressure: q.
Exposure factor: Ce.
Gust effect factor: Cg.
External pressure coefficient: Cp.
How to calculate the lateral load of wind.

In the consistent shells, because of consistency, the shear force will easily be transferred to the structural supports, but in a reticulated shell, for having control over sheer force, it is better to add diagonal elements to the network. That is why we chose the triangulated meshing at the first.

The next step is to determine the cross-section of beams. As in shell structures with their very little thickness, our first structural purpose is to minimize the bending force, and as the bending force has an inverse relation with the moment of inertia, so we chose the pipe-shape cross-section for the beams.

Since one of our purpose in the optimization step is optimizing the structure meanwhile minimizing the structural weight, we considered the radius of the pipes as the third variable, in a way that avoiding waste of materials in the optimization and form-finding level, we could find the best dimensions for our possible cross-sections. The common materials used in these structures are wood and steel, and considering the climatic conditions of the region, we introduced steel as our material to Karamba3D.

After the pre-processing step, we will get to the processing step. In fact, after introducing the structural details in the former step, in this step, we will calculate the finite element. During the third step that is named post-processing, we begin to interpret the results. In this step, Karamba3D presents some pieces of information, including weight, displacements of nodes, stresses, etc. in numerical and graphical forms to the user (Fig. 3). We will use this numerical information in the optimization step to realize the objectives of optimizing.

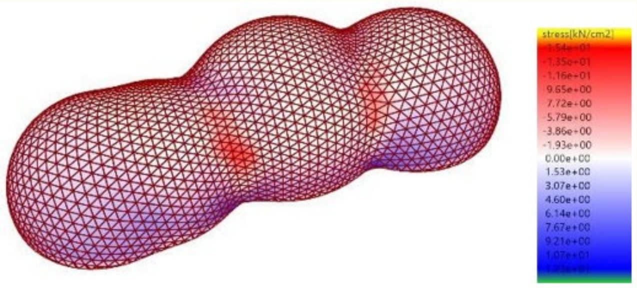


Figure 3: Graphical Interpretation of the Primary Instance, after calculations regarding the finite element.

Penalty function

Before starting the optimization step, for defining other intended architectural criteria and avoiding the generation of unreasonable forms, we will put some penalty functions in the middle of the optimization process. In the first step of the research, after generating the primary form, we introduced a rectangular area to the computer, and it began to generate the points in this allowed area. As a result, the occupancy area of the produced forms would have remained between "0" to the whole of the defined rectangular area. On the one hand, one of our objectives in the optimization step was to minimize weight as much as was possible. Hence, the computer will automatically place the points on the closest possible situation to each other, and in the latest produced forms, the occupancy area would be almost "0". This was not acceptable for us, so considering the allocated physical program, we introduced the 6000 m2 as the minimum occupancy area to the computer. During penalty function1. We defined an algorithm that forced all the generated forms to have an area more than the defined one. Indeed, the confinement of occupancy level became between 6000 m2 and the whole of the determined rectangular area. Any produced form with the occupancy level corresponding with this range would be acceptable for us.

Still, we had another important problem to consider. Although we should have generated a consistent volume, there was a possibility for producing some dispersed volumes that in sum cover the minimum area of 6000 m2. And that was not acceptable for us. Hence, during Penalty function 2, we made the computer give us a consistent volume. Otherwise, the penalty function would be applied.

As we told during the first step, according to the physical program, we decided to have four spatial zones. Despite defining the minimum acceptable area, there is a possibility for producing a form that according to Fig. 4, has the acceptable occupancy area, but due to the short distances between the center of spheres in it, we could not separate the four zones.

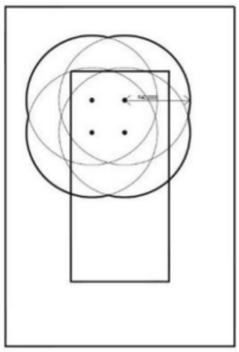


Figure 4: An example of unacceptable results.

For avoiding the production of such unacceptable forms, we compared the distances between the points with the radius of spheres drawn from those points. To choose two of the four points, we had six different choices. Therefore, with six separate functions, we forced the produced spheres to have more center-to-center distance than their radiuses in sum (Fig. 5). So, the spheres will be tangent to one another or acceptably far from each other. And that is an agreeable situation.

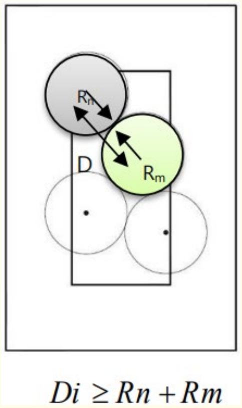


Figure 5: The Rule of Penalty function.

Optimization

Optimization for realizing the minimum cost and energy consumption, and achieving the maximum output profit and performance, is one of the most significant questions regarding structure (Yang. 1970). Generally, we have three kinds of structural optimization: 1- optimization in size 2- optimization in shape 3-optimization in topology (Pham. 2016). In the first kind of optimization, the general shape of the structure is determined, and we merely optimize the dimensions of the structure. In the second kind, the undefined thing is the shape or the layout of the structure. And in the third kind that we are using in this research, forces, conditions of structural supports, and other restrictions imposed by the architect are determined, and the aim is to arrange the structure fundamentally (Bendsoe and Sigmund. 2013).

Although in any proposed case, there are infinite numbers of possible answers, the optimization algorithms will help the user to find the best one. In an optimization problem, any objective function has its own special characteristics in terms of easiness to be minimized or maximized (Akbulut et al. 2020). Genetic algorithms are kind of evolutionary algorithms that use biological theories based on concepts such as heredity, mutation, and natural selection to find an optimized formula for predicting or adopting the best patterns and finding the best answers. These algorithms are one of the most potent multi-purpose optimization algorithms. Using this algorithm and the Wallacei plug- in in the environment of Grasshopper, we will produce optimized forms based on structural performances. Actually, the outputs of the evaluation process will merge with the primary geometric model and start the production and modification of more geometrical models based on the purposes of optimization (Fig. 6). At the end of this process, the architect will have many optimized formal instances and have the right to choose between them.

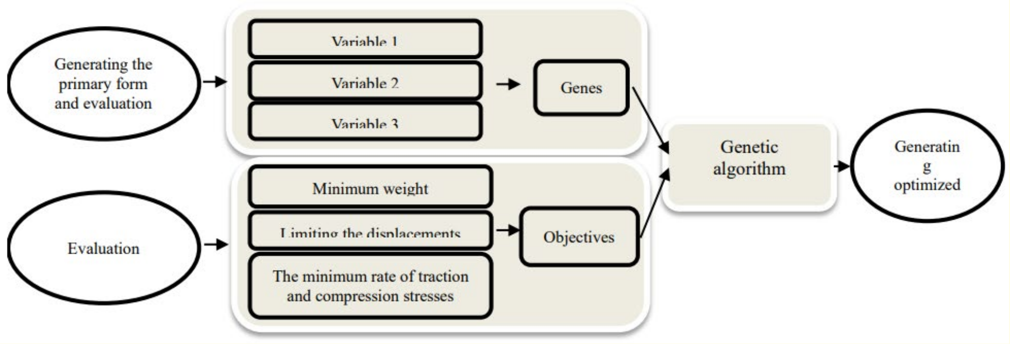
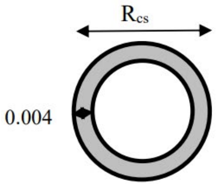


Figure 6: Optimization flowchart.

The mathematical functions used in the optimization process are including:

The weight function

In the building industry, lightweight structures are of immense importance, as the weight of structures affects the structural performance so much. Besides, a major part of construction costs spends on the structure, and using fewer materials could make our building financially more efficient. Reticulated shell structures are lightweight and therefore financially beneficial. Optimization of these structures has made them one of the best alternatives for covering spans greater than usual. The case for current research is a reticulated shell with triangulated networks made of steel pipes that the triangulated area between them would be filled with ultimately lightweight materials. The thickness of pipes is considered fixed, and their cross-section diameter would input to the algorithm, as the variable3. Considering this variable in calculating the weight of elements and also the influence of variables1 and 2 on changing the topology of structure and the total weight of the structure, the generated forms will be appraised, and regarding our objectives, the most optimized formal alternatives will be produced. Eq(7).



$$\begin{gathered}\rho = m/v \\ \rho = 7850\ kg/m2 \\ v = \pi\ (R2 - r2)\ h \\ W\ (x, y) = ((\pi(x)\ 2-(x-0.008)2)\ {*}y)\ {*}7850 \\ x = Rcs \qquad 0.1m < Rcs < 0.2m \\ y = L \qquad 1m < L < 3m\end{gathered} \tag{7}$$

The weight function for each of the elements

The displacement function

Shells are very delicate structures, so we should consider the issue of buckling a significant one in their design (Lefevre et al. 2015) Consequently, the structure should not be subject to large displacement. On the other hand, the more flexible structures have better performances enduring lateral forces. Hence, the rate of displacements should be limited in a determined spectrum. Evaluating all the parameters with the help of finite element, the Karamba3D plug-in will show the user how much the structure should be displaced. One of the major purposes of optimization is to limit the rate of this displacement in a spectrum less than 12 centimeters (D max< 0.12 m) (Grande et al. 2018) According to Eq(5), the size of structural displacements has a direct relationship with imposed forces and an inverse one with the stiffness of the structure. So, considering our objective to decrease the size of this displacement, we should increase the structural stiffness. The stiffness of these elements is dependent on their material and subject to their lengths and cross-section diameters Eq(8). The stiffness of the whole system and hence, the displacement of the whole relies on the stiffness of each element in the topology of the structure (variables 2 and 3).

$$
\begin{gathered}
\epsilon = \frac{\delta}{l} \\
\sigma = E\epsilon \rightarrow \sigma = E\frac{\delta}{l} \\
\sigma = \frac{F}{A} \rightarrow F = \sigma A \\
F = E\frac{A}{l}.\delta \\
[F] = [K].[D] \\
K = E\frac{A}{l} \\
K_1(x,y) = \frac{E\pi(x)^2}{y} \\
x = R_{cs} \\
y = L
\end{gathered}
$$

Stiffness Function for Each of the Elements

The axial stresses function

In shells, the most optimized form is the one in which the elements transfer the loads through the axial stresses that consists of traction and compression stresses (σ = T+P). In this research, after evaluating the finite element in the second step, we will find the rate of axial stresses in each of these elements. The Karamba3D plug-in will show the compression stresses with negative amounts and tractions with positive ones. The less an element could tolerate a stress, either in the form of compression or traction, the less material it will need to have an optimized design. Therefore, minimizing the total amount of these stresses will be considered as one of the objectives of optimization.

Since we wanted to minimize the objectives of the supposed project, putting some penalties in the way of the optimization process, we forced the computer to inhibit the minimizing and to multiply the objective functions in a large number, if our intended architectural conditions were not realized (penalty functions). In this way, after producing inappropriate volumes, the genetic algorithms will consider these answers as children with undesirable characteristics and eliminate them from the evolution process. Accordingly, during this process, their Genes will be modified toward becoming a minimized objective function, and as a result, we would have acceptable answers.

During this loop when we reach to a point at which our answers in continues generations are close to each other, the form-finding process will be stopped.

Findings

The proposed algorithm was scripted and executed in the environment of the grasshopper according to the abovementioned method. Solving possible errors required multiple back-and-forth and dealing with detailed errors. Optimization in this research had 100 generations and in each generation 50 instances. After the optimization, the existing instances in the first level of the Pareto front could have been chosen as the acceptable answers, because of their closeness to the answer of the problem. Comparing the first-generation instances to the middle and last generations showed that the expected amounts in the objective function had improved gradually (Fig.7). As we could see in Fig.8, the integral of the graph had shrunk during the process; and the closer we got to the center of the graph, the more symmetric became the distribution of variables. In this way, we could say that the last generation encompasses the closest answers to this question. It means that the best forms (according to the intended changes) and the best structures (according to relevant variables) are realized in the last generation. Also, studying the results of the objective function in some random examples of the last generation showed that there were not any significant differences between these examples. And the designer could have chosen each of these 50 examples of 99 generations as the final answer. Choosing between these 50 examples could be dependent on other design parameters such as the designer`s aesthetic preferences, and its process is out of the scope of this research.

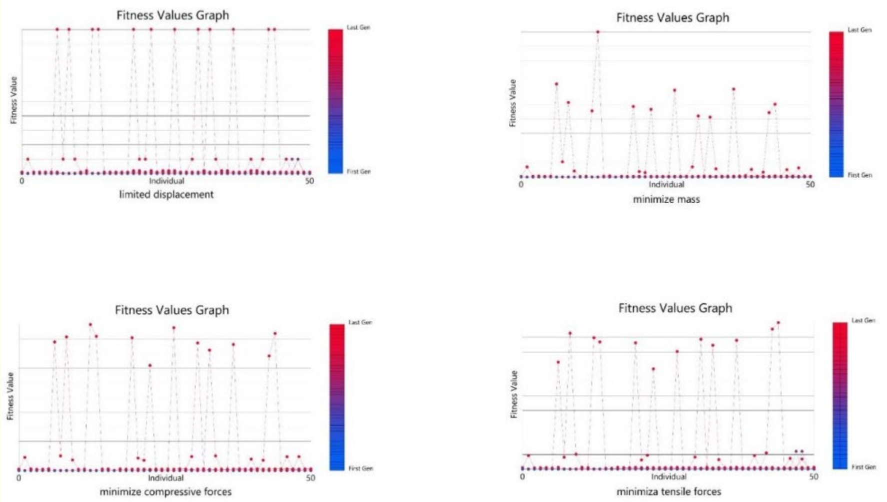


Figure 7: Graph of the objective functions.

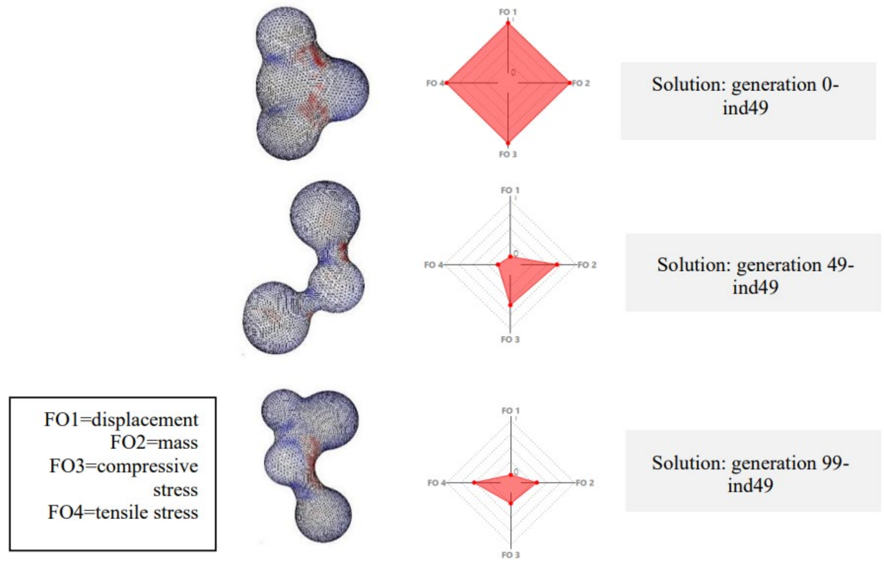


Figure 8: Graph of the objective functions.

Conclusion

Until now, most of the form-finding and structural design methods were merely based on civil engineers' knowledge, and the architect`s role was not so bold, so often the designed structure and the architect`s intended form had too many differences. The research presented here, with introducing a three steps method in the environment of the grasshopper shows a new way for form- finding in shell structures which could be analyzed completely by the architect. In this method, considering the structural elements, structural support conditions, loads, cross-sections, and the used materials, the weaknesses of the structure are analyzed with the finite element with the help of the Karamba3D plug-in. Finally, using the Wallacei plug-in, the weight, axial stresses, and structural displacements were introduced as the optimization functions. At the beginning, special characteristics of the project such as the area, the size of intended spaces, and the physical program were inputted into the process. Although we used this method in regard to the abovementioned parameters, and our case was restricted by some structural and architectural criteria, replacing the special features of each project, we could generalize these algorithms to other form-finding problems in shells as our findings highlight the correctness of the proposed method and its potential to be used as a tool for architects.

References

1. Adriaenssens S., et al. Shell Structures for Architecture: Form Finding and Optimization, Routledge, Abingdon, Oxfordshire, England, UK (2014).
2. Akbulut M, Sarac A and Ertas AH. "An investigation of non-linear optimization methods on composite structures under vibration and buckling loads". Advances in Computational Design 5.3 (2020): 209-231.
3. Bletzinger KU and Ramm E. "Form finding of shells by structural optimization". Engineering with computers 9.1 (1993): 27-35.
4. Bucalem ML and Bathe KJ. "Finite element analysis of shell structures". Archives of Computational Methods in Engineering 4.1 (1997): 3-61.
5. Du Peloux L., et al. "Construction of a large composite gridshell structure: a lightweight structure made with pultruded glass fibre reinforced polymer tubes". Structural Engineering International 26.2 (2016): 160-167.
6. Du Peloux L., et al. "From shape to shell: a design tool to materialize freeform shapes using gridshell structures". Design Modelling Symposium Berlin, Berlin, Germany (2013).
7. Fedorik F., et al. "Use of design optimization techniques in solving typical structural engineering related design optimization problems". Structural Engineering and Mechanics 55.6 (2015): 1121-1137.
8. Grande E, Imbimbo M and Tomei V. "Structural optimization of grid shells: Design parameters and combined strategies". Journal of Architectural Engineering 24.1 (2018): 04017027.
9. Hussain M., et al. "On mixing the Rayleigh-Ritz formulation with Hankel's function for vibration of fluid-filled functionally graded cylindrical shell". Advances in Computational Design 5.4 (2020): 363-380.
10. Jiang Y. "Free form finding of grid shell structures". M.S. Dissertation, University of Illinois at Urbana-Champaign, Champaign, USA (2015).
11. Kaveh A and Bakhshpoori T. "An efficient multi-objective cuckoo search algorithm for design optimization". Advances in Computational Design 1.1 (2016): 87-103.
12. Kilian Az and Ochsendorf J. "Particle-spring systems for structural form finding". Journal of the International Association for Shell and Spatial Structures 46.2 (2005): 77-84.
13. Lefevre B, Douthe C and Baverel O. "Buckling of elastic gridshells". Journal of the International Association for shell and spatial structures 56.3 (2015): 153-171.
14. Liddell J. "Frei Otto and the development of gridshells". Case Studies in Structural Engineering 4 (2015): 39-49.
15. Loukaides EG and Seffen KA. "Multistable grid and honeycomb shells". International Journal of Solids and Structures 59 (2015): 46-57.
16. Malek SR. "The effect of geometry and topology on the mechanics of grid shells". Ph.D. Dissertation, Massachusetts Institute of

Technology, Cambridge, Massachusetts, USA (2012).

17. Oxman R. “Performance-based design: current practices and research issues”. International journal of architectural computing 6.1 (2008): 1-17.
18. Paoli CCA. “Past and future of grid shell structures”. M.Eng Dissertation, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA (2007).
19. Pham AH. “Discrete optimal sizing of truss using adaptive directional differential evolution”. Advances in Computational Design 1.3 (2016): 275-296.
20. Pillwein S., et al. “On elastic geodesic grids and their planar to spatial deployment”. arXiv preprint arXiv:2007.00201, ACM Transactions on Graphics 39.4 (2020): 125.
21. Tang G. “Timber gridshells: beyond the drawing board”. Proceedings of the Institution of Civil Engineers-Construction Materials 166.6 (2013): 390-402.
22. Tedeschi A and Andreani S. AAD, Algorithms-aided Design: Parametric Strategies Using Grasshopper, Le Penseur Publisher, Brienza, Potenza, Italy (2014).
23. Winslow P, Pellegrino S and Sharma SB. “Multi-objective optimization of free-form grid structures”. Structural and Multidisciplinary Optimization 40.1 (2010): 257-269.