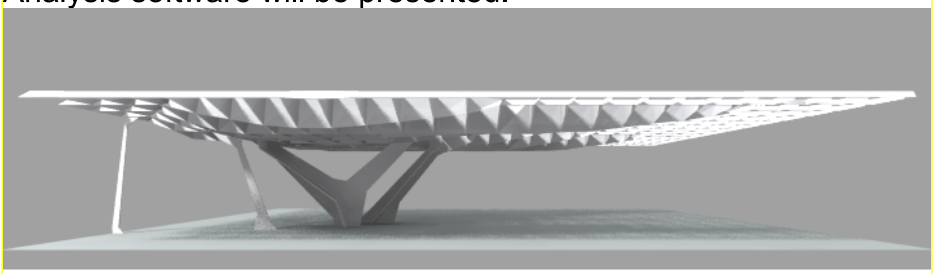
Ioannis Chatzikonstantinou

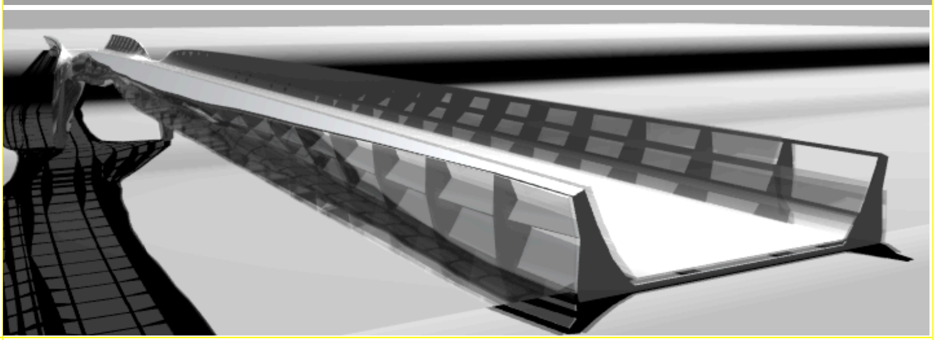
Paper : **Structural Optimisation as a means for exploring design alternatives.**

Topic: Architecture

Authors:
Ioannis Chatzikonstantinou
Greece
www.volatileprototypes.com

Abstract:
In architectural design the utilisation of optimisation methods alongside the designers own process of decision-making can be advantageous in many cases. Generative tools based on optimisation algorithms can become a meaningful source of design inspiration, offering variations which stem from compliance to specified performance parameters. Optimisation-based generation and transformation of form can therefore lead to an efficient design through an explorative design process.
It is common in architecture that non-quantifiable aspects of a design are important and therefore need to be maintained through any optimisation process. In other words, forms produced by designers need not always be optimal performance-wise; on the other hand, they include unquantifiable traits irrelevant of any performance factor that still need to be maintained.
Given the potential of optimisation-based generative and transformative methods, this study will focus on the development of an optimisation environment suitable for the architectural design process, that relies on simple mathematical principles and can produce design variations rapidly, conforming to design traits. The tool that has been developed operates on arbitrary surfaces supplied by the user in the form of a polygonal mesh and produces an enriched egg-crate or dual-surface (sandwich) representation of the structure considering parameters of structural efficiency such as stiffness. The output of the process can be exported for further processing. Case studies as well as evaluations of the tool's efficiency by commercial Finite Element Analysis software will be presented.





www.generativeart.com

Contact:
yconst@gmail.com

Keywords:
design performance, structural optimisation, shell structure, eggcrate-structure, generative tool

Computational Optimization as means for exploring structurally efficient surfaces

Ioannis Chatzikonstantinou, Architect
www.volatileprototypes.com
e-mail: yconst@gmail.com

1. **Abstract**

In architectural design the utilization of optimization methods alongside the designers own process of decision-making can be advantageous in many cases. Generative tools based on optimization algorithms can become a meaningful source of design inspiration, offering variations which stem from compliance to specified performance parameters. Optimization-based generation and transformation of form can therefore lead to an efficient design through an explorative design process.
A design often exhibits non-quantifiable aspects which need to be maintained through any optimization process. Forms produced by designers need not always be optimal performance-wise; on the other hand, they often include unquantifiable traits irrelevant of any performance factor that are of value.
Given the potential of optimization-based generative methods, this study will focus on a computation-based form-finding method that aims to maximize the structural efficiency of user defined surfaces in 3-dimensional space. The overall form of a surface is maintained and it's thickness distribution is optimized to achieve structural efficiency. The method relies on simple mathematical principles and can produce design variations rapidly.
A digital tool that implements the proposed method has been developed as part of this study. The tool operates on arbitrary surfaces supplied by the user in the form of a polygonal mesh and produces an enriched egg-crate or dual-surface (sandwich) representation of the surface geometry considering parameters of structural efficiency such as stiffness. The output of the process can be exported for further use. Case studies as well as evaluations of the tool's efficiency by commercial Finite Element Analysis software (TNO Diana) will be presented.

Keywords

design performance, structural optimization, shell structure, eggcrate-structure, generative tool

2. Introduction

During the last decade, there has been an increasing interest in the field of architecture for the application of algorithms and methods that generate forms and structures based on constraints and parameters defined by the designer. Such methods are of special interest in that they add an additional 'exploratory' dimension to design: The definition of forms, geometries and features is transformed from a process of generating to a process of conducting 'form-finding' functions. In these cases, the designer becomes beneficially alienated from the project, inheriting the role of a regulator rather than a generator of architectural geometry and form. Matters of personal pref-

erence can therefore be expressed through the control of process parameters, rather than by direct manipulation of physical features, an approach that may prove beneficial as it maintains methodological integrity.
When generative methods are designed based on performance factors, so as to converge to solutions that are as close as possible to the optimum regarding these factors, an additional degree of significance is attached to the generated results. Optimized results produced by such generative processes constitute a meaningful design exploration that can significantly enhance the design's performance in parallel to producing innovative and unexpected solutions.
The need for architectural form-finding applications can be found mostly during the early stages of the design process, where design decisions have a profound impact on the geometry and form of the building in a large-scale. In this design phase, a quick overview of alternatives and efficient communication of ideas and solutions are essential.

3. Background

Optimization of designs for physical objects by the use of computers is a multidisciplinary research field that has gained significant attention during the past thirty years. Structural Optimization is a field of engineering that has seen a tremendous benefit during the last thirty years, with the introduction and increasing application of computational processes in engineering. Many finite element based algorithms have matured and have been implemented into software packages applicable to day-to-day practical problems and often implement optimization processes that take advantage of computational analyses. Indeed, there has been during the last decade an unforeseen increase in the amount of literature related to Computation-based Optimization. One of the most well known methods for structural optimization has been proposed by Xie et al. [7] in 1992 and is known as Evolutionary Structural Optimization (ESO). Topology Optimization is another similar family of optimization methods, pioneered by Bendsøe and Kikuchi in 1988 [6]. Topology Optimization was the first method that allowed unconstrained variety in the resulting forms.
In architectural design, automated and computation-based processes began having an influence during the last decade. When used in an architectural context, optimization processes are present not as a finite process that eventually leads to an optimal or near-optimal result, but rather as an interactive process that is based on the conversation between the designer and the computer. This relationship is outlined by Miranda in his study titled Self-Design and Ontogenetic Evolution [4]:

"[...] The problem is anyway approached not with the intention of finding optimal solutions, but challenging and creative ones. It is not answers the computer should provide, but questions about the problematic of the design. It is in this context of "problem- worrying" (as opposed to problem solving) that the work has been carried."

An optimization or form-finding method in an architectural design context becomes then an integral part of the design process and acts as a mediator between the project and the designer.

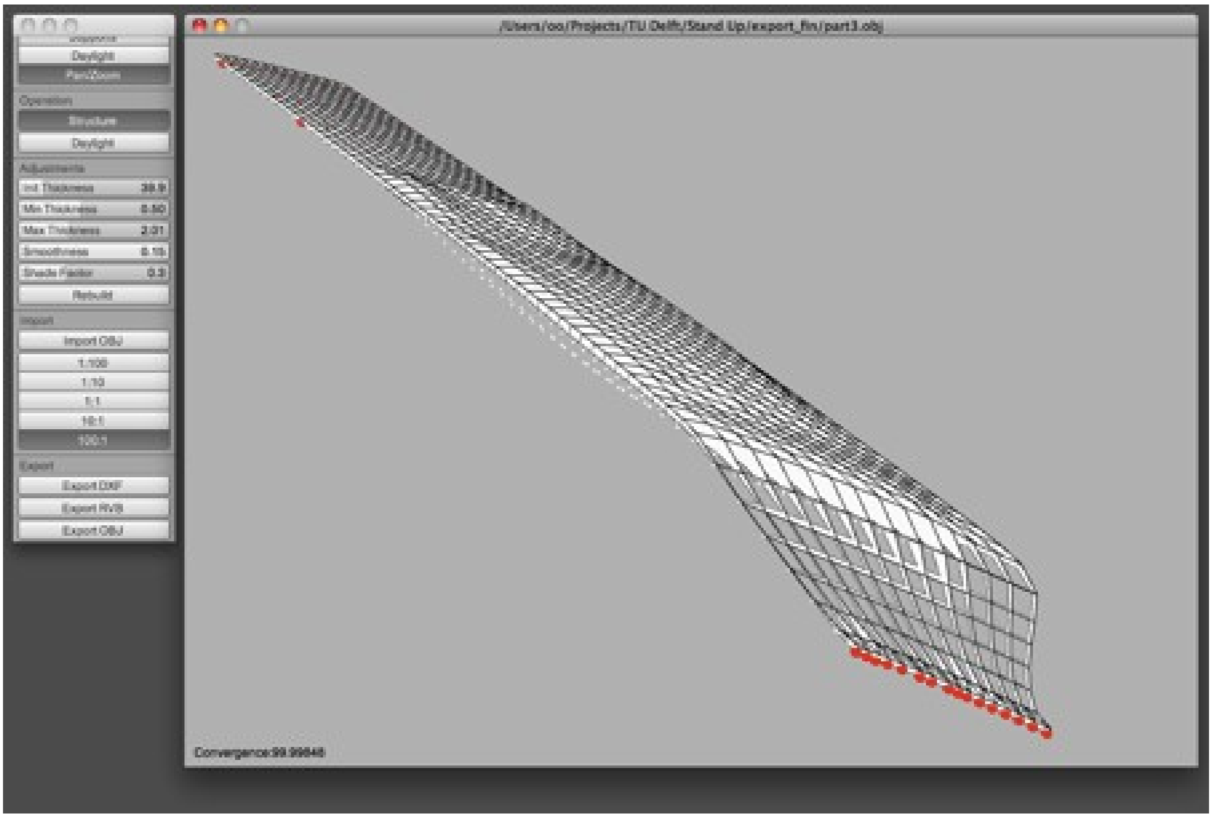


Fig. 1: User interface of the digital tool implementing the proposed method.

4. Optimization Method

The proposed method is a form-finding algorithm inspired by structural optimization techniques and aiming to maximize the structural rigidity of an arbitrary, user-defined surface geometry by finding an optimal distribution for the surface's thickness, with the goal of achieving maximum structure lightness. It primarily functions by varying the local thickness of each region of the user-specified surface, according to local loading and deflection conditions. In addition to the method, it's implementation into a digital design/form-finding tool will be discussed. The user interface of the tool can be seen in Figure 1. It has been developed using the Java programming language and the Processing classes.

4.1. Principles

Considering the elastic deformation of a uniform element, the relationship between stress and strain is:

$$\sigma = E \cdot \varepsilon \qquad (1)$$

where:
σ : Stress.
E : Elastic Modulus.
ε : Strain.

In a beam with a given height and under moment load, the stress on a point of a section is given by:

$$\sigma=\frac{M}{I_x}\cdot y \quad (2)$$

where
M : Moment applied to beam.
Ix : Second Moment of Area around the x-axis.
y : Perpendicular distance to centroidal x-axis.

It can be derived from (2) that increasing the Second moment of area (I) decreases the stress for a given point on a beam section. For simple cross-sections (e.g. rectangular), the second moment of area is given by:

$$I_x=\frac{bh^3}{12} \quad (3)$$

where
b: width in x-dimension
h: height

Combining (1), (2) and (3):

$$\varepsilon_h=12\frac{M}{E\cdot b\cdot h^2} \quad (4)$$

Which demonstrates that an increase in height of a cross-section (thickness in the case of a surface) yields an overall decrease in strain around the section. In order to counter-act deflections, an increase in thickness of a structure in areas with the most stress is required. However, since with the change of geometry the distribution of stresses also changes, the ideal case is to perform optimization in an iterative analysis-transfer cycle.

4.2. Process

The steps followed during a single run of the form-finding process are the following:

A surface mesh is imported.
A lattice, truss-like structure is derived by it.
The boundary conditions are set by the user.
The optimization process is carried out in an iterative way: In a continuous loop, the structure is analyzed and then optimized.
The resulting structure/mesh can be exported for further use.

While this process may be in principle similar to the one implemented by optimization methods used in the engineering fields, the design intention is quite different. Architectural design is a process where high reciprocity can be observed between the designer and the design environment. Therefore it was a primary intention to create a process that allows interactivity. In contrast to optimization and form finding methods used in the fields of engineering, the method in question adds the dimension of design exploration, allowing the user to change parameters in real time and making it easy to restart/pause and resume the form-finding process.

4.2.1. Geometry Input/Output

A key aspect of the proposed optimization algorithm is to be able to act on user-defined geometry: The program accepts as input arbitrary surfaces supplied by the user in the form of a polygon mesh. This geometry therefore becomes the ‘starting point’ for any operation.
The structure that is eventually participating in the optimization process is a lattice representation of the original surface. The topology of the input mesh serves to set-up a graph of interconnected nodes, which is in turn offset by an initial, user-defined value along the surface’s normal vectors. This value represents the thickness of the surface, and each of it’s local values is therefore the object of the structural optimization algorithm. The lattice structure is derived by generating straight and diagonal elements between neighboring points, which in turn form truss-like elements. The topology and connectivity of the input mesh defines therefore the geometry of the generated structure.

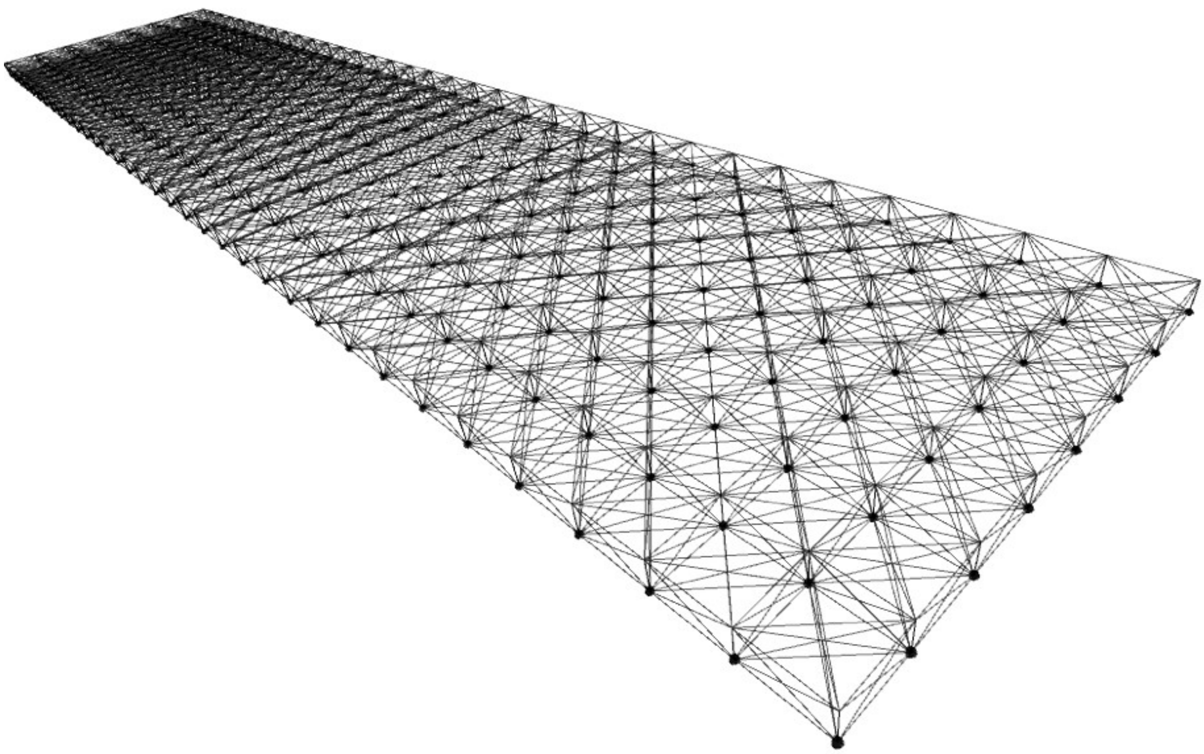


Fig. 2: The lattice structure used for optimization.

The lattice that is generated for a flat rectangular surface can be seen in Figure 2. The exact connections that happen during the generation of the lattice structure can be visualized in Figure 3. If the quad element [A1-A2-A3-A4], seen in this figure, is

part of the imported mesh, then the actions that are taken to translate it into a lattice structure are the following:
An equivalent quad [B1,B2,B3,B4] is generated by offsetting the original.
Each of the points of the original is connected (apart from the original connections) to four other neighboring points. So, for example, A1 would be connected to A3, B1, B2, B3 and B4. The importing of mesh definitions in Wavefront OBJ format is already implemented and expansion to include other formats is planned.
After the optimization process has finished, the exported geometry is derived by replacing the lattice structure with quad mesh elements. The ‘ribs’ perpendicular to the surfaces and two sets of surface quads are exported in different object groups for ease of manipulation.
Export is possible in DXF, OBJ and RVB (Rhinoscript) format.

4.2.2. Structural Analysis

The structural analysis is carried out on the lattice structure that is the result of the surface import process. It is performed using a simple Dynamic Relaxation loop on the set of points and members of the structure, regarding the members as stiff springs and concentrating the mass of the structure on the points. The deformation of the springs is then used as an indicator of the strain at that particular region of the surface.
Dynamic relaxation is an iterative process in which a system reaches equilibrium by following a pseudo-dynamic process in time, each iteration based on an update of the geometry [3]. Dynamic Relaxation techniques and point-spring models are frequently used to predict the outcome of real-time physics-based applications. One such method, similar to the one used in this case, has been described by Jakobsen [2]. One of the main advantages of iterative relaxation techniques is that they offer scalable real-time performance, so that accuracy can be traded for speed of execution. However, the accuracy of such a method is by no means comparable to analysis algorithms such as Finite Element Analysis, mainly because of the simplified structural model used. Nevertheless, the need for such functionality in this case is not in order to substitute dedicated analysis methods, but rather to produce a rough estimate of the stresses on the structure in order for optimization to take place.

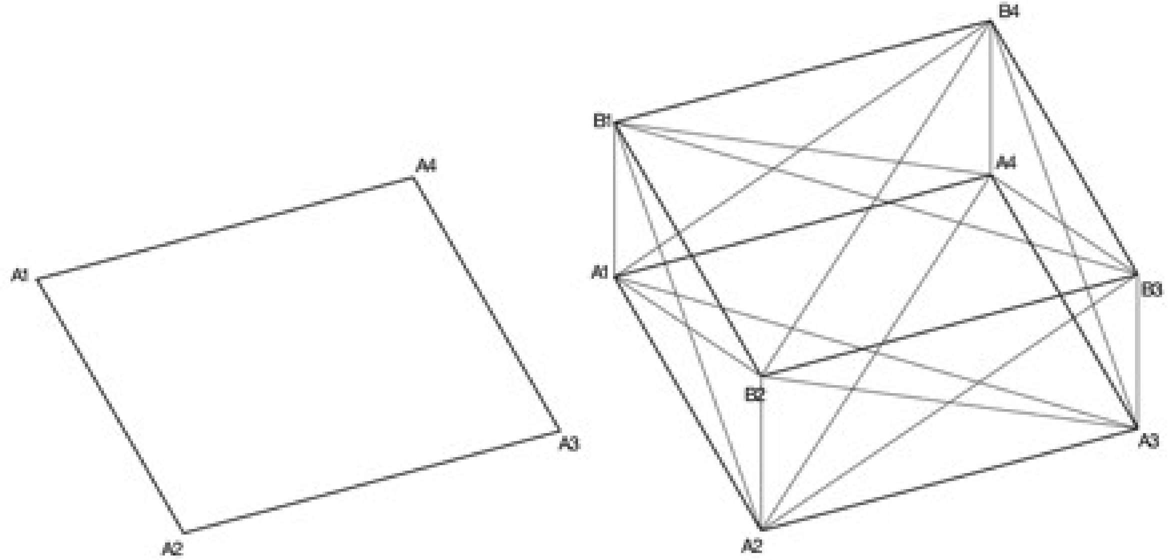


Fig. 3: a. Initial Surface quad, b. Connectivity of a single quad element of the lattice structure.

4.2.3. Structural Optimization

Optimization is performed after each analysis step. The optimization process functions primarily by re-distributing the thickness between regions of the user-specified surface, based on the local strain they receive. The nodes whose neighboring members experience the least amount of strain, transfer a fraction of the thickness of their adjacent members perpendicular to the primary surface, to a global 'buffer' pool, which is in turn used as a 'source' to increase the thickness of the perpendicular members with the most stresses. It is a process based partially on the concepts behind Evolutionary Structural Optimization, the main difference being that the domain of solutions here is the thickness combinations that are possible for a given surface.
The exact parameters that define the process can be set before or during the optimization process. Namely, it is allowed to control high and low limits to the thickness of the surface, as well as a 'smoothness' factor, which specifies how intense the exchange of thickness values is between the various regions of the surface. The manipulation of these values allows a certain control over the process and therefore variety of resulting geometries.
The time-based nature of the analysis and optimization process introduces some peculiarities to the method. One important point in the process is that, in order to determine the new equilibrium position after an analysis-optimization loop, the equilibrium of the immediately previous step int time is used as a starting point for the dynamic relaxation, instead of starting every time from the initial condition. This offers a significant speed-up with apparently no significant loss in accuracy. This assumption is based on the fact that, for a time-based relaxation method and purely elastic deformation, a structure will converge to equilibrium from any starting point, given enough iterations of the process. Therefore, if the previous equilibrium is close enough, a fewer number of steps are required to reach the current one. In order to further ensure that optimization is only performed when the analysis is close to equilibrium, the amount of thickness transfer is penalized according to the amount of convergence of the structural analysis.

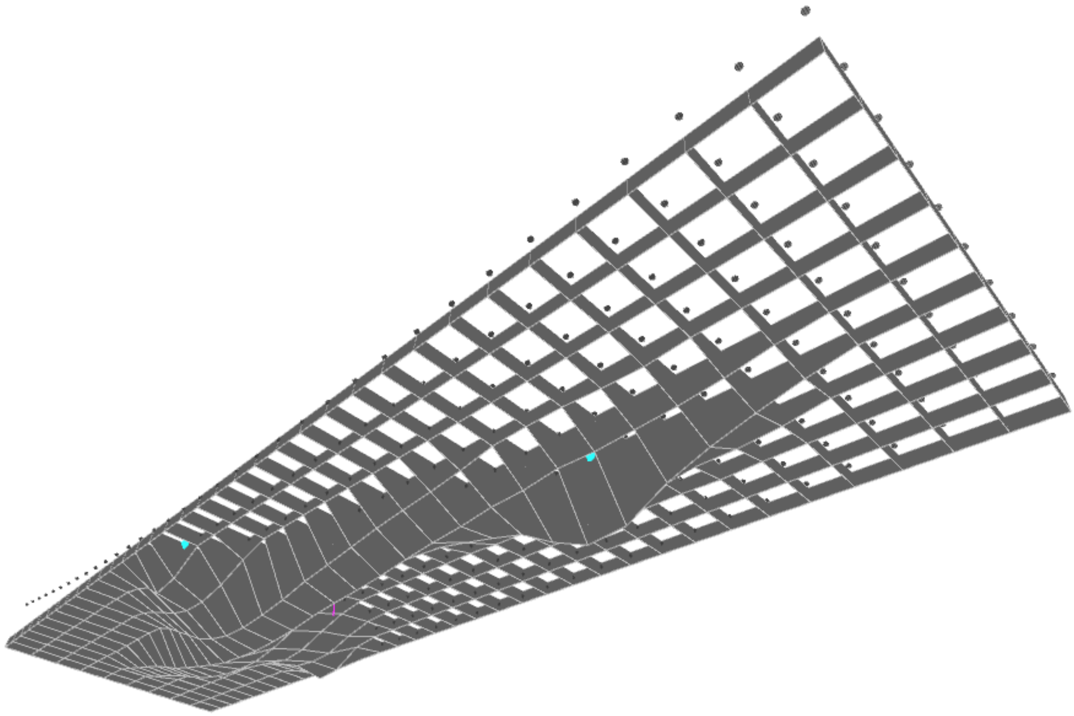


Fig. 4: A planar surface after optimizing with respect to a certain arrangement of supports.

5. Evaluation and Performance Assessment

5.1. Structural Evaluation

In order to evaluate whether the form-finding process indeed is meaningful and adds structural efficiency, a simple test case was formulated and tested in Finite Elements Analysis (FEA) software. The test consists of a simple mesh being optimized by the method in question and then tested by FEA against the original, non-optimized form. The analysis mesh consisted of a ribbed, 'egg-crate' structure, with the form of a flat rectangular surface. Supports were specified in three positions close to the edge of the surface. The support conditions can be seen in Figure 5. For the mesh elements, the type CT30F was chosen. It is a triangular element, therefore two of them were used for each quad. For the material properties, a set that reflects the elasticity, Poisson ratio and density of structural steel was chosen. This can be seen in Table 1. Both original and optimized structures, along with their respective displacement diagrams can be seen in Figure 6. The calculations were performed in TNO Diana 9.3.
Based on the Finite Element analysis it can be argued that the optimized version of the geometry tested indeed demonstrates a better material distribution than the original. Results indicate that the maximum displacement has been reduced by about 65% (original: 81mm, optimized: 28mm), and principal stresses are reduced by about 40-50% and are more evenly distributed, as a result of the increased thickness close

to the points with the highest stress. The structural optimization performed by the method in question can be characterized as satisfactory in terms of performance, at least for simple cases like the one presented.

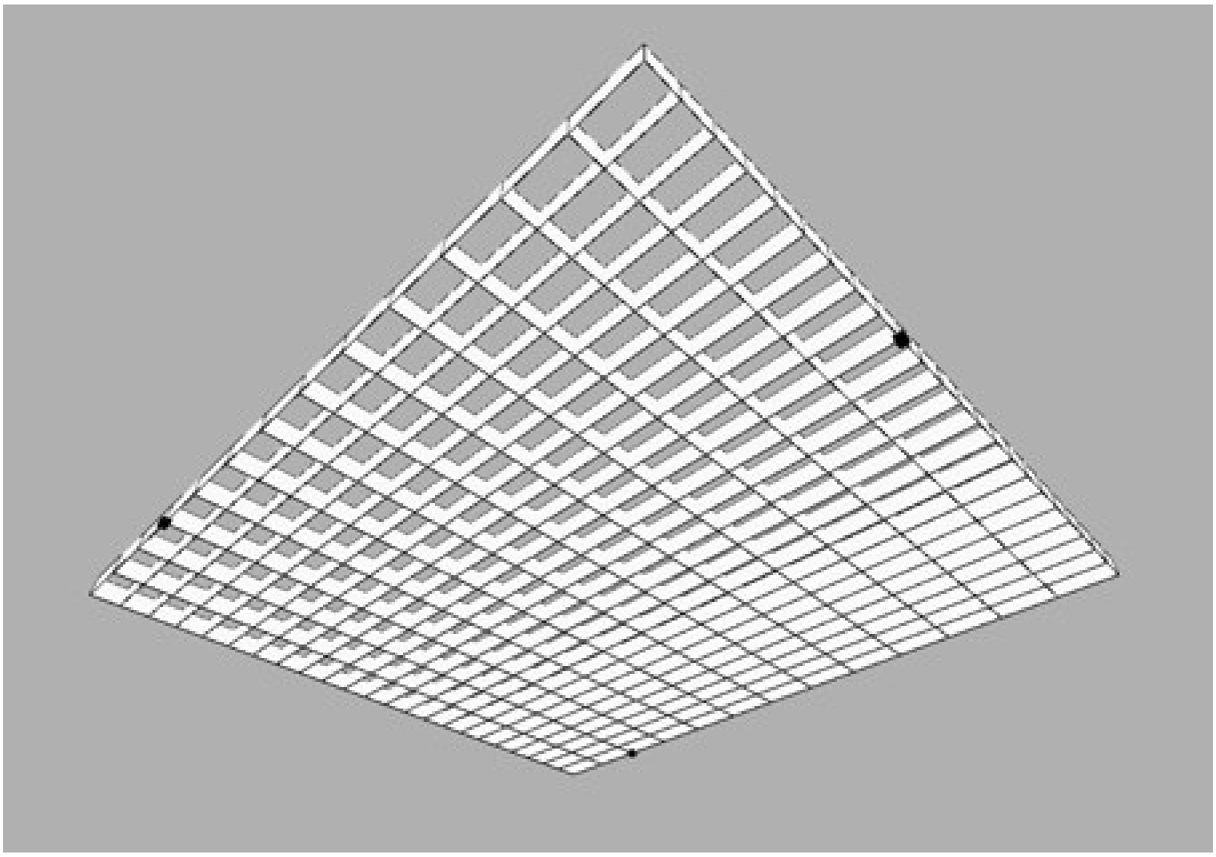


Fig. 5: Support conditions for the FEA.

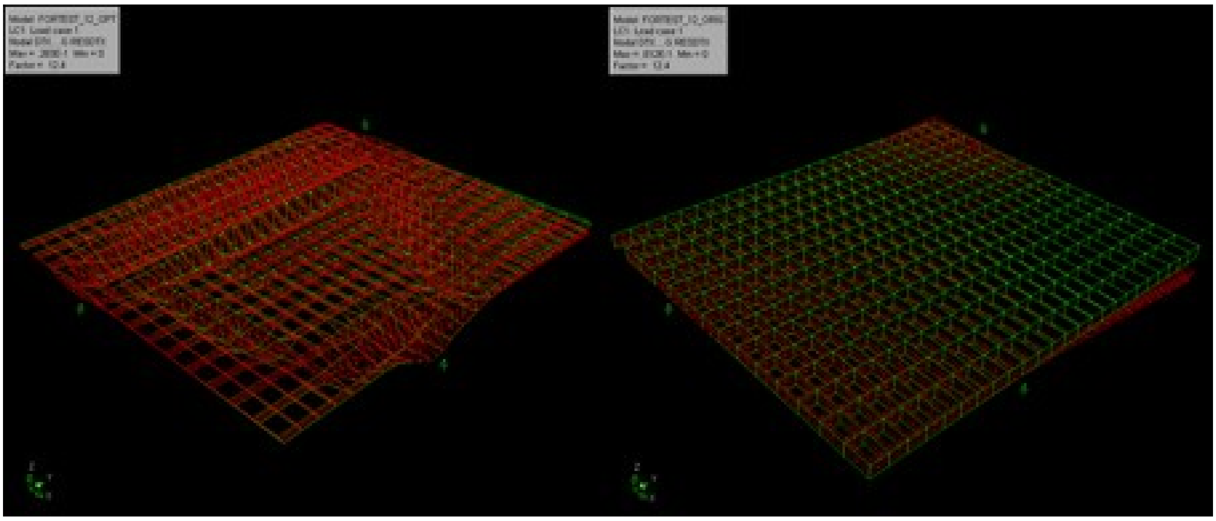


Fig. 6: Displacement diagram of the optimized (left) and original (right) geometries. Produced by TNO Diana 9.3.

5.2. Symmetry Test

A simple test that can be used to simply determine if an optimized result is indeed sub-optimal, is the symmetry test: On a mesh with one or more symmetry axes, sup-

ports are laid out such that symmetry is maintained. Whether the resulting material distribution maintains the symmetry of the original, is considered an indication of the correctness of the optimization, since ideally optimized structures based on symmetric meshes are indeed symmetric.
In this case, rectangular surfaces were tested, after symmetrically placed supports were defined on them. The results can be seen in Figures 7 and 8. While they may not resemble the absolute optimum solution, symmetry is maintained.

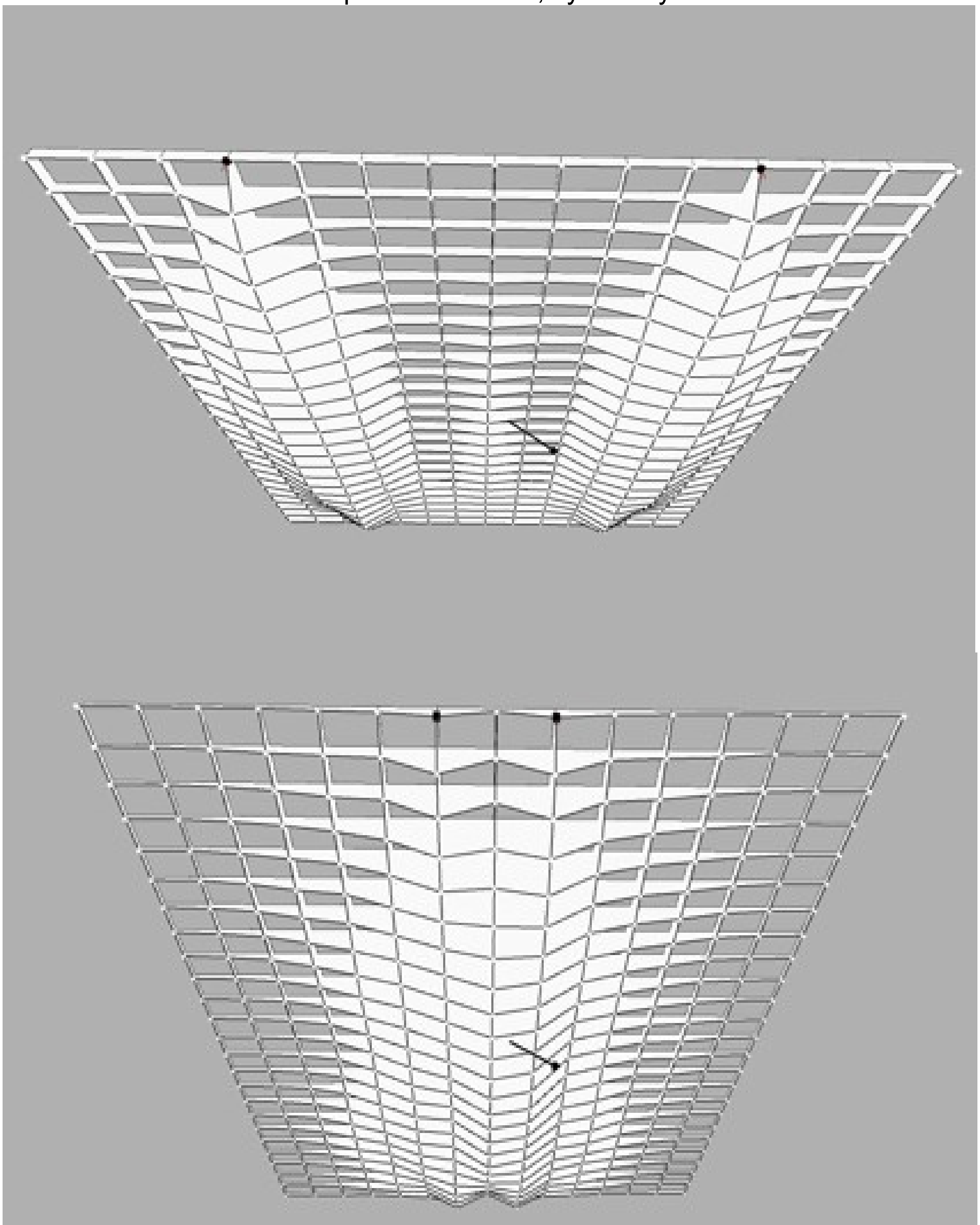


Fig. Fig. 7,8: Two instances of symmetry tests on a planar surface.

6. Architectural Case Studies

Following performance evaluation, a few case studies were carried out in order to determine and evaluate the qualitative traits of the resulting geometries. Two case studies will be presented:

A pedestrian bridge
A rectangular shelter

The geometry of the first case study has been generated in the Rhino 3D modeler, out of the sweep of a single curve along a straight axis, with a number of intermediate cross-section curves. The geometry folds at about one third of the length of the bridge in order to form supports. The initial surface can be seen in Figure 9, the resulting surface can be seen in Figures 10 and 11 and the relevant egg-crate structure in Figure 12. In Figure 12 especially, the features introduced by the optimization process can be observed in the variation of the sizes of the members.
The surface was meshed by Rhino and imported as an .OBJ file into the optimization tool, and consists exclusively of quad elements. Supports were specified at the two edges and at the folded members. Optimization was carried out using a relatively high smoothing factor.

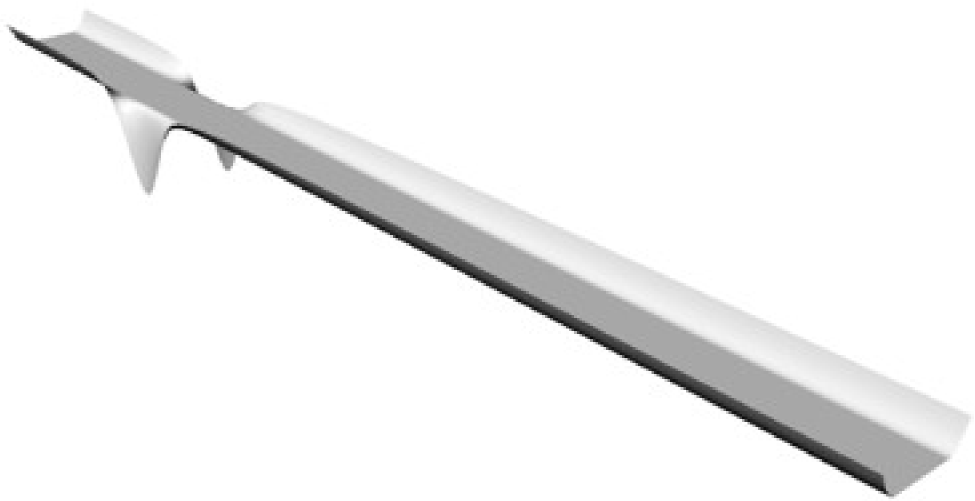
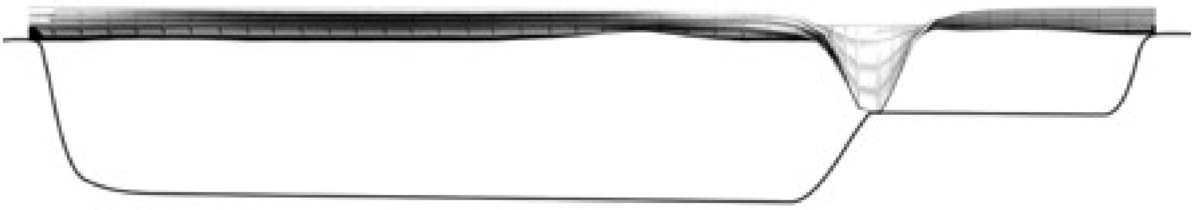


Fig. 9: Initial surface of the first case study: a pedestrian bridge

F



ig. 10: Side view of the pedestrian bridge.

Fig. 11: Perspective representation of the pedestrian bridge.

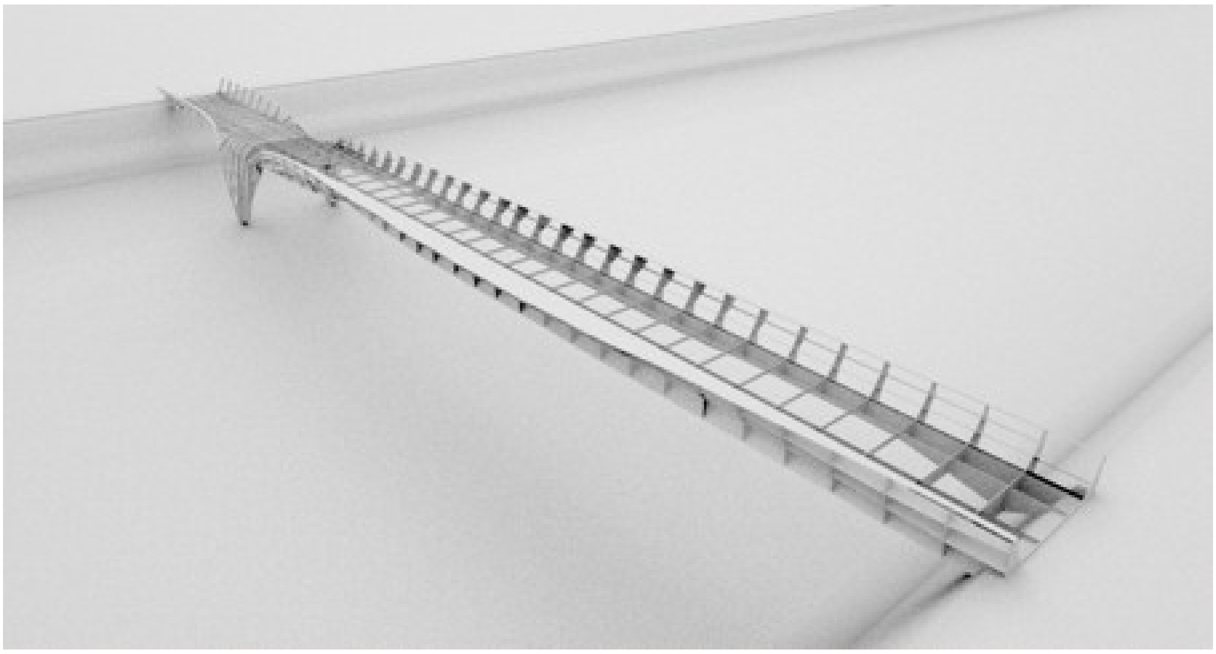


Fig. 12: Perspective representation of the pedestrian bridge's egg-crate structure.

The second case study, the rectangular shelter, was derived from a simple rectangular planar surface, subdivided into a number of quad elements. After import, 3 supports were specified near the center of the surface. The resulting 'egg-crate' structure can be seen in Figures 13 and 14.
Judging from the case studies it could be argued that the produced structure is indeed expressive of it's function, adding to the surface a formal dimension related to structural efficiency. As already mentioned, the intensity of this expressive effect can be controlled before and during the optimization process, by altering the process parameters.

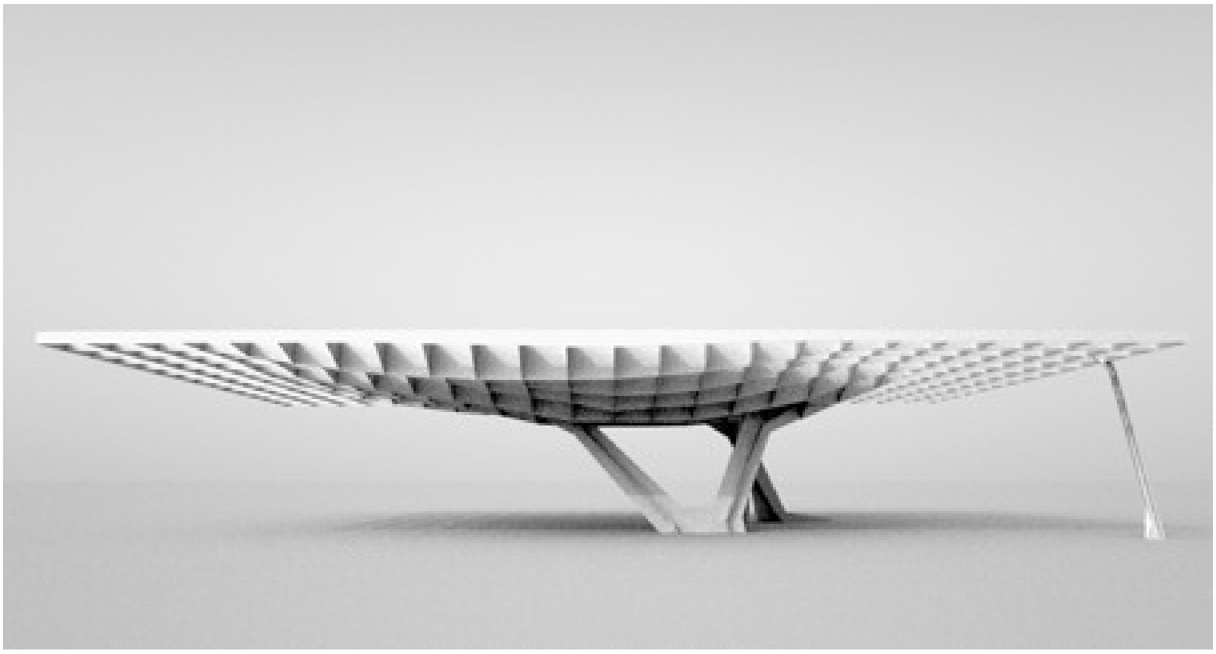


Fig. 13: Perspective representation of the secons case study: A rectangular shelter. Underlying supports have been assed manually.

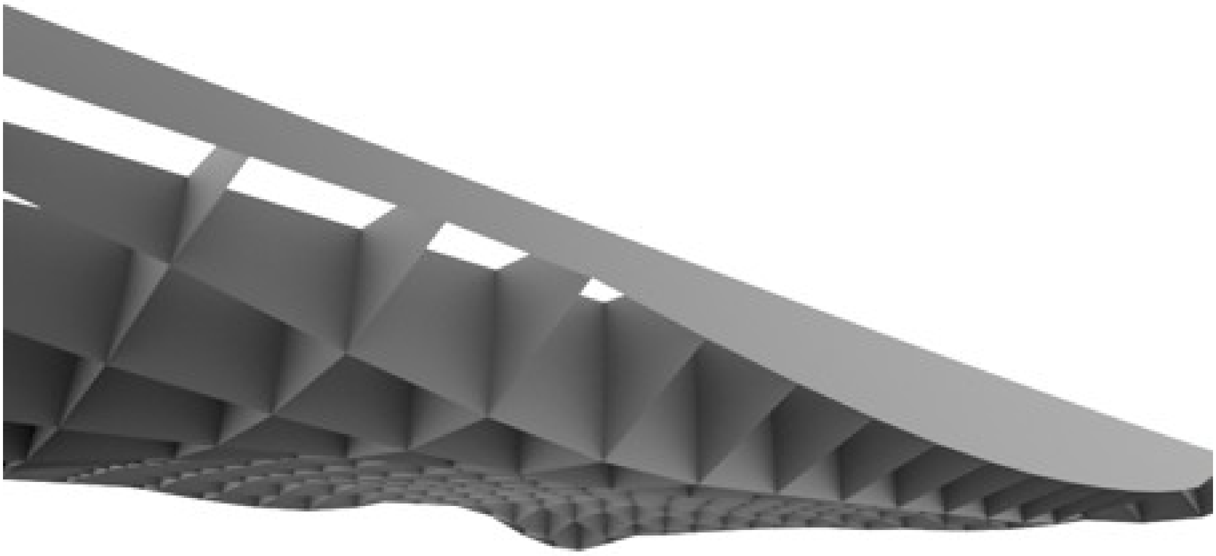


Fig. 14: Perspective representation revealing the variation in thickness of the rectangular shelter.

7. Discussion

The results from the FEA analyses seem encouraging and demonstrate that indeed the use of a dynamic relaxation method for simple analysis of lattice structures, such as the one used here, is feasible.However, it should be noted that, while efficient for basic geometries and simple in implementation, the structural analysis method that is implemented is rather crude and may easily yield inaccurate results. The implementation of a proper Finite Element Solver could significantly increase the credibility of the structural analysis. This could be achieved in two possible ways. First, an interface with a commercial FEA product or library and second, the implementation of a simple in-house FEA algorithm. However, there is a question of whether this would impede real-time performance, affecting the interactive nature of the tool.

Due to the translation of the input mesh to a specific structural form ('egg-crate' structure), a specific tectonic strain is introduced to the otherwise abstract input geometry. Therefore the output of the program is more specific than the input. While this may be beneficial in some cases where e.g. quick visualization of structure is in favor over the application of a user-defined construction scheme, there is still a lot of room for re-thinking and improvement, in order to improve the variety of applications of the tool and allow for easier integration with the design process.

A significant improvement related to data exchange would be the implementation of NURBS surface import/export. Then, instead of using a mesh to generate the underlying lattice structure, the control points of a NURBS surface could be used. This shift would improve drastically the tool's exchange capabilities, which would in turn aid it's integration with the design process. Taking this functionality a step further, an interface with 3D modeling applications could even be considered, such as Rhino or Grasshopper, allowing direct manipulation of user geometry.

Given the real-time operation of all functions in the program, one application that could be considered and would be extremely interesting would be the simulation of kinetic/adaptable structures that respond to changes in loading conditions. The existing programmatic implementation could be combined with a physical prototype or even a tectonic system that would allow a variety of responsive objects to be produced, objects that change their geometrical characteristics depending on loading conditions.

8. Conclusion

In this study, the development of a structural optimization method targeted at generating efficient surface structures, and it's implementation into a digital tool have been discussed.
The tool can perform structurally efficient optimizations of material distribution on user-supplied surface geometries. The optimization process that has been implemented is based on a simple Dynamic Relaxation model and favors variation and real-time performance over accuracy and reliability. Initial evaluations of the method's efficiency in terms of structural performance are encouraging. In this sense, the method is already suitable for use where formal exploration based on structural function is desired.
The geometries that come as a result of the optimization process exhibit expressive traits relevant to their structural function. Some of the aspects of these traits can be controlled before and during the optimization process, allowing an extra degree of freedom to the designer and maintaining the necessary amount of interaction between the user and the process.
The study that has been discussed in this paper is by no means complete. The author is actively engaged in extending the tool's functionality and improving it's performance.

9. **References**

[1] Hofmann, A., Bollinger, K. Grohmann ,M., Generating Geometry of Irregular Frameworks Algorithmically, in Proceedings of the First Symposium on Architectural Geometry Vienna, Austria September 13-16, (2008)

[2] Jakobsen, T., Advanced Character Physics, presented in the Game Developers Conference, (2001)

[3] Lewis, W. J., Tension Structures: Form and Behavior, London, Telford (2003)

[4] Miranda, P., Self-Design and Ontogenetic Evolution, in Proceedings of the 4th Generative Art International Conference, (2001)

[5] Serway, R., Physics for Scientists and Engineers, 6th edition, Brooks Cole, (2003)

[6] Schramm, U., Zhou, M., Recent Developments in the Commercial Implementation of Topology Optimization., in IUTAM Symposium on Topological Design Optimization of Structures, Machines and Materials, (2006)

[7] Xie, Y. M., Steven, G. P. Evolutionary structural optimization, Springer-Verlag, Berlin (1997)