Proceedings of the Fifteenth International Conference on
Computational Structures Technology
Edited by: P. Iványi, J. Kruis and B.H.V. Topping
Civil-Comp Conferences, Volume 9, Paper 4.8
Civil-Comp Press, Edinburgh, United Kingdom, 2024
ISSN: 2753-3239, doi: 10.4203/ccc.9.4.8

Fitness Criteria for the Optimization of Load-Bearing Structures in Comparison

C. Müller[1] and K. Deix[2]

[1]bewegende Architektur e.U., Vienna, Austria
[2]Institute of Material Technology, Building Physics and Building Ecology, Vienna University of Technology, Vienna, Austria

Abstract

Structural optimization is not just about reducing material and increasing load-bearing capacity. Architecture has concepts that allow an ideal use of the building, maximise the quality of stay, react to sights and much more. How can these goals be negotiated? Which concepts are in conflict and which complement each other? How can synergy effects be created? In this paper, we examine the interweaving of architectural concepts and structural design using a fictitious example of a bridge. We used our in-house developed software "Phänotyp", a software specially designed for form- and cross-section optimization in the field of architecture. The process is carried out step by step and the different optimization methods are compared

Keywords: gradient descent, genetic algorithm, architectural, structural optimization, open source, finite element analysis, phänotyp, blender3D, PyNite

1 Introduction

The software "Phänotyp" is based on the finite element library "PyNite" [2], which is integrated in the 3D graphics software Blender3d [3]. This is a free 3D graphics software that is used in the field of DCC (Digital Content Creation) for film and game development. In "Phänotyp" it is used as a GUI (Graphical User Interface), whereby the ease of use and the diverse possibilities for designing and modifying objects are particularly important. Since Blender3d can be controlled with Python code, the tool was written in Python also and can be installed as an add-on [4] in Blender3d crossplattform. It has already been used frequently at the Vienna University of Technology in the design process for teaching architects. A key function the software offers - compared to conventional finite element software - is the ability to optimise both, the member cross-sections and the entire structural shape. Several algorithms are available for the cross-section optimization of the members, which when applied iteratively, lead to an ideal utilisation of the profiles. Genetic algorithms and a gradient descent method are implemented to optimise the entire structure. These solve the multidimensional optimization problem that arises using the fitness functions.

In this paper we examine the interweaving of architectural concepts and structural design. It should be noted that no formula is given in this article, instead the planning process for architects should be documented. We took a fictional example which is shown in Figure 1. It is a bridge with a span of 20 m. The first idea is the variation of the middle section (sk1). The wider it is, the more space is created on the bridge for people to linger. On the other hand, minimising the area would be a much smaller load case. The second concept is to vary the height (sk2). Both the arc shape and the suspended geometry will most likely result in a higher performance of the structure. But is it comfortable to walk up and down for everyone? The third concept - as an architectural element - is to align the bridge with a landmark (sk3). Visitors should ideally lead visually towards this destination. The last concept is to vary the height of the parapet (sk4). A higher parapet height is expected to improve the performance of the structure. A low parapet maximises the view of the landmark. We would like to avoid having the parapet at the height of the eye right? How can we negotiate this concept to a feasible solution?

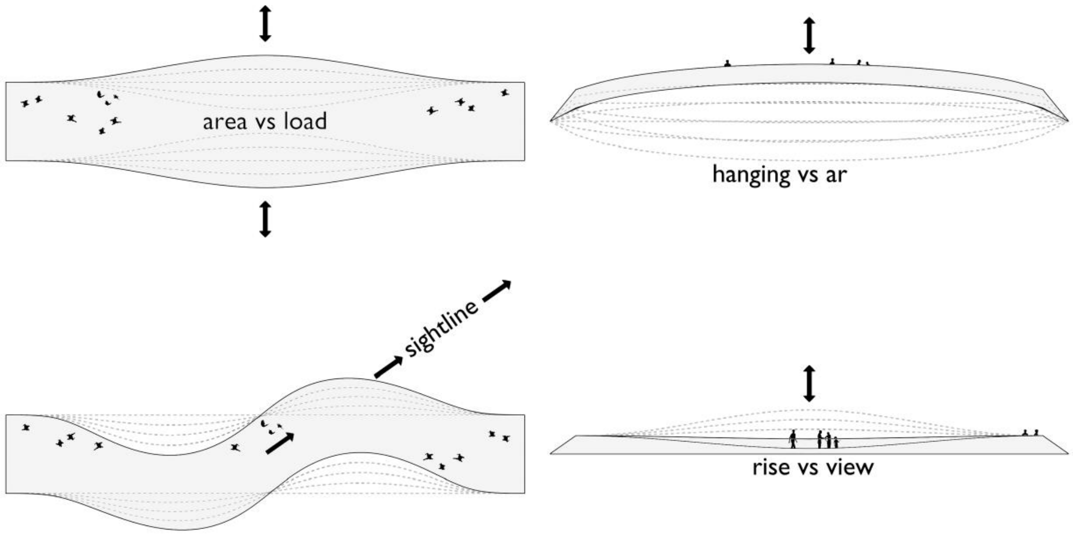

Figure 1: Architectural and structural concepts.

2 Definition of shape-keys

In the second part, our fictitious bridge structure is defined in so called “shape keys”. A shape-key (sk) is a method to store a movement of a set of vertices within the geometry. In this example the shape-key always varies from 0 to 1. In order to create two directions for a shape-key, the starting geometry is deformed in the first direction at position 0. The second direction of transformation is available at position 1 of the shape-key. All possible variants between these two stages are morphed automatically. Figure 2 shows the variation of the bridge in form of “shape-keys”, where each line shows the shapes described above in minimum, medium and maximum formations.

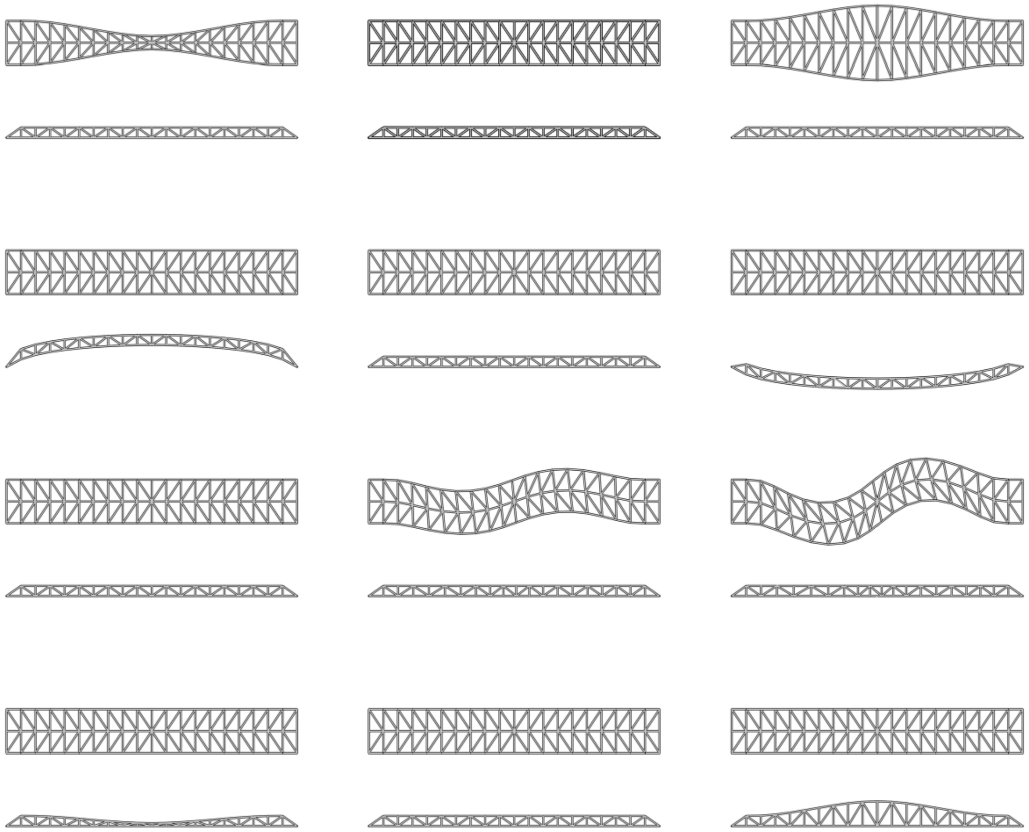

Figure 2: Definition of morphology in floor plans and sections (1. row: sk1, 2.row: sk2, 3.row: sk3 and 4.row. sk4)

In Figure 3 the shape-keys are shown also in 3D. Furthermore it is possible to morph these keys among each other. Let's take a look at [0.0, 0.5, 1.0, 0.0] for example. These four keys indicate the values of each shape-key. The first key set to 0.0 will result in the maximal tapering of the structure. The second key set to 0.5 will result in a straight section without any arc or hanging trough. The third key set to 1.0 will result in the highest twist towards the sight. The last key set to 0.0 will lead to a minimal parapet available in this range of variations. Figure 4 shows the variations.

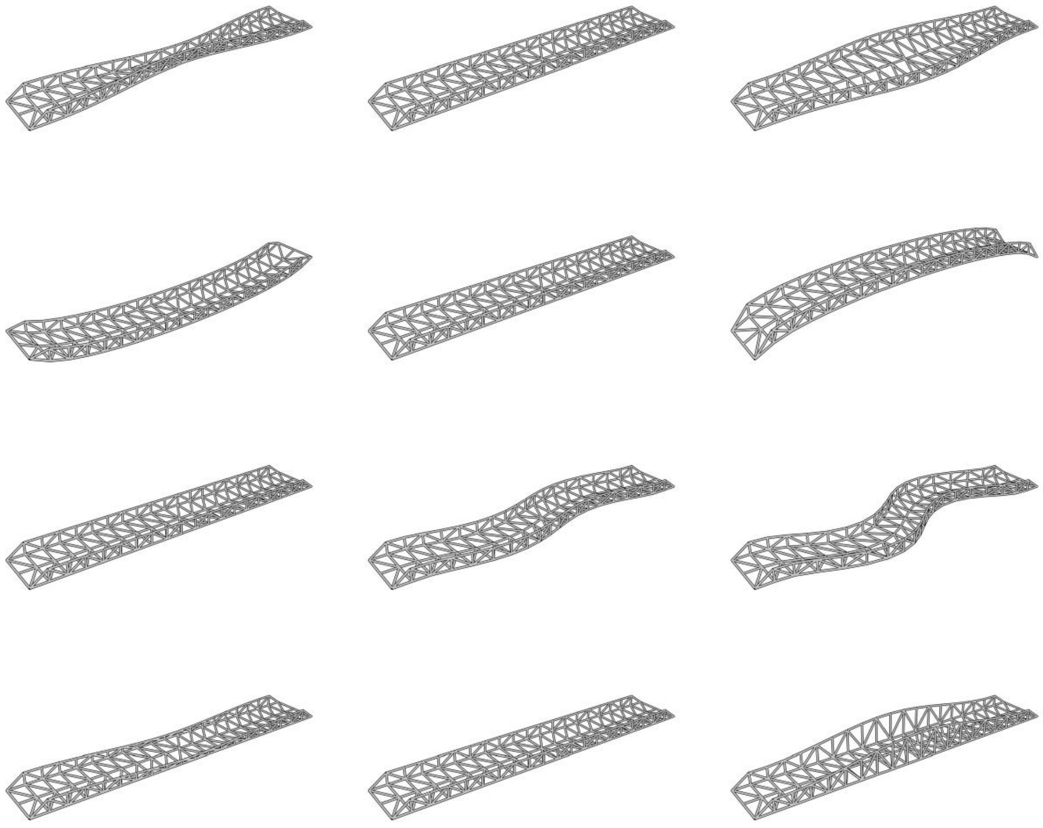

Figure 3: Definition of morphology in 3D.

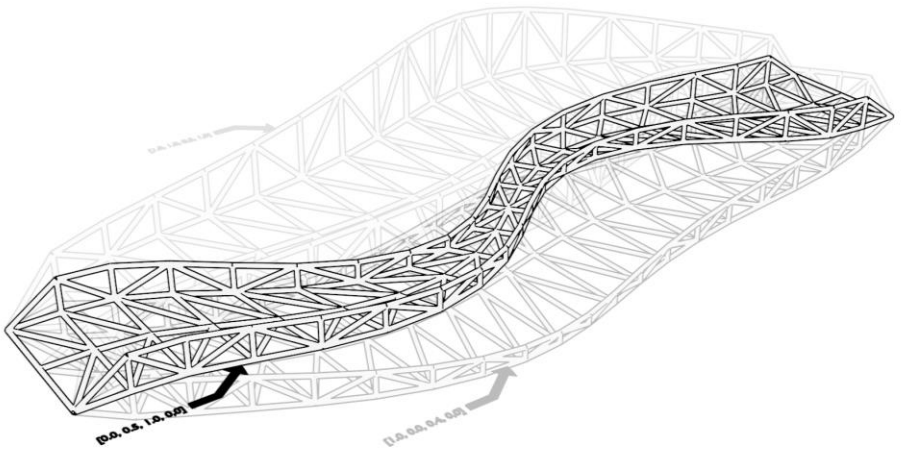

Figure 4: Variation of shape-keys for three examples.

3 Break down into dual opposing variants

In order to make the comparisons easy to read, we will concentrate here on comparing two keys in each case (Figure 5). The aim is to provide designers and planners with an overview of the consequences of the formal intervention in the form of the construction. With three confronted keys, there are 6 possible variants. For these 6 combinations two variants were considered in which the bridge is either movable (one one side) or fixed (on both sides). This creates a matrix with a total of 12 shapes of which 121 variants each are calculated. As a fitness function the minimal strain energy was used, so low values of the fitness are better. The results of the calculations are shown in Fig.6 and Table 1.

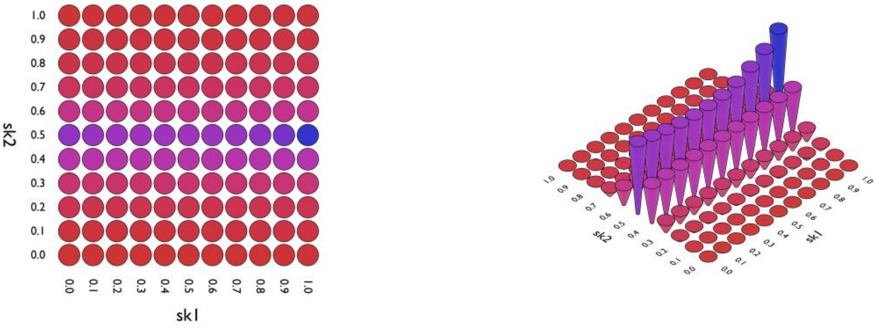

Figure 5: Fitness-values with the shape-key 1 and 2.

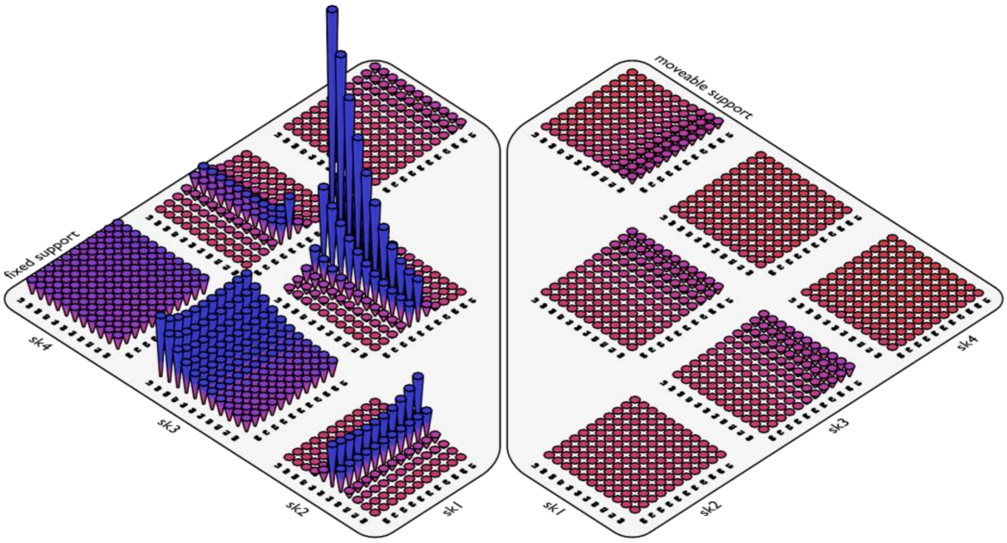

Figure 6: Comparison of all pairs with fixed and movable supports in isometric.

Combinations of shape-keys	highest fitness for fixed support	highest fitness for movable support

sk1 | sk2	4.105	0.137
sk1 | sk3	4.519	0.727
sk1 | sk4	1.181	0.131
sk2 | sk3	19.081	0.526
sk2 | sk4	3.138	0.106
sk3 | sk4	0.493	0.547

Table 1: Fitness (strain energy) of the 12 variants.

It can be seen clearly that the hanging, as well as the arch form, performs very well. The worst part is the straight section of the bridge. It can also be seen that the fixed support results in greater variation in the fitness function, whereas the one-sided sliding support results in relatively low (= favourable) and uniform fitness values, which means that the influence of the shape is not so great. Especially the combination of sk1 with sk2 and sk3 as well as sk2 with sk3 show strong influences of the strain energy in the case of fixed support.

4 Gradient descent and genetic algorithm

To calculate every combination (in Phänotyp labelled as "Bruteforce") requires a lot of computing power. With two shape-keys there are $11^2 = 121$ variants, with 3 there are $11^3 = 1331$, with 4 there are $11^2 = 14641$. In this section we extend the solution of the problem with two further methods of "Phänotyp". With gradient descent (GD) and genetic algorithm (GA) we can achieve the same result much faster. These solve the multidimensional optimization problem that arises using the fitness functions. Several fitness functions are implemented, such as the longitudinal stresses, strain energy, deflection, total weight, enclosed volume, etc. These can also be combined and weighted with each other. The overall shape can be changed along defined node paths using the "Proportional Editing" function in Blender. This results in different shape-keys and allows an optimised building shape to be found that fulfils the desired fitness functions.

Figure 7 shows the result of the calculation with the genetic algorithm [5]. To make the results better readable, only every 6th image is shown. The results are sorted from best to worst fitness.

With "Phänotyp" it is als possible to perform a sectional optimization. Over-utilized cross-sections of the beams are enlarged and under-utilised cross-sections are decreased. Iterative application results in optimised cross-sections for all members, whereby a 3 to 5-fold optimization is sufficient for simple structures. Figure 8 shows the results of a GA calculation of a 5-fold cross-section optimization as an example.

The results in the form of the fitness function for the minimum distortion energy are shown in Figure 9 (GA) and 10 (GD). The different solution paths of these different methods can be clearly recognized.

Figure 7: Genetic algorithm sorted by fitness with every 6 results. From best to worst (top left to bottom right).

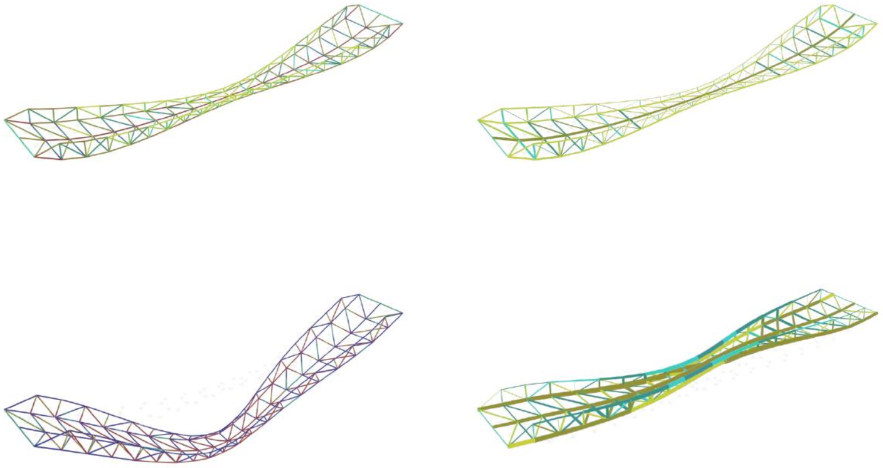

Figure 8: Comparison of the GA calculation a 5-fold cross-section optimization, from best to worst (top left to bottom right).

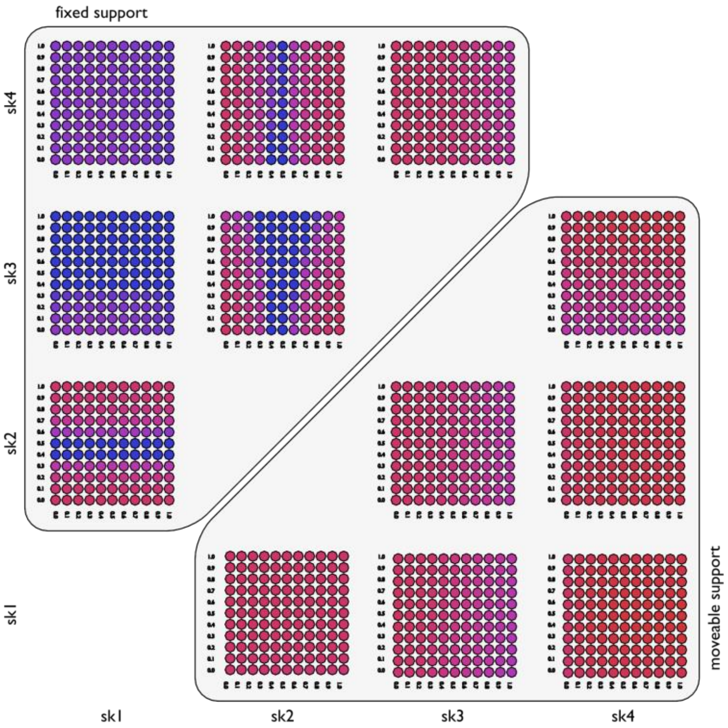

Figure 9: Fitness-values for GA

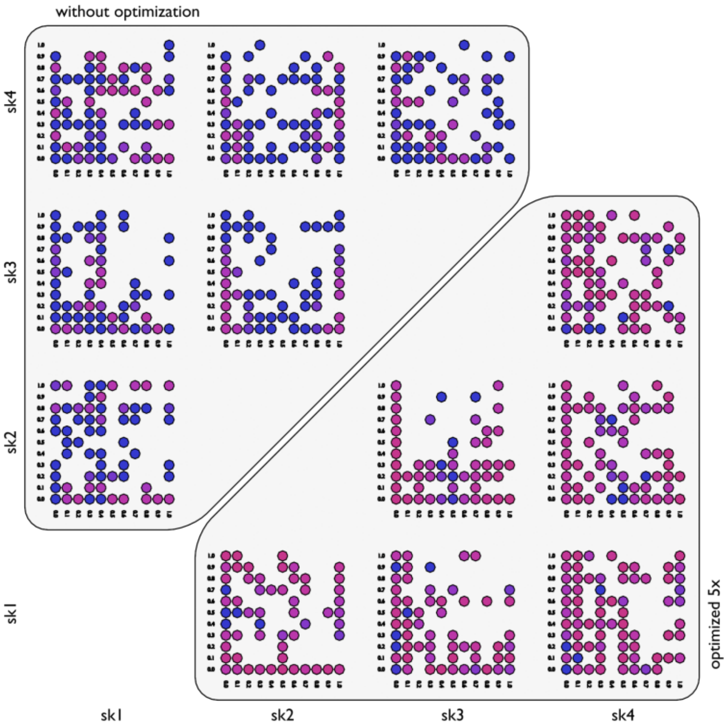

Figure 10: Fitness-values for GD

Table 2 shows the final result in the form of the optimum values for shape keys. It can be seen that in all cases, regardless of the support, a uniform width of the bridge (sk1 = 0.5), a curved elevation (sk2 = 1), a straight bridge (sk3 = 0) and a slightly higher parapet (sk4 = 0.3 to 0.6) are favourable.

	sk1	sk2	sk3	sk4
GA	0.5	1.0	0.0	0.6
GD	0,418	0,998	0,012	0,351
GA 5x opt	0.5	1.0	0.0	0.3
GD 5x opt	0,457	0.992	0.0	0.337

Table 2: Optimised shape-keys (fitness: minimum of strain energy)

The different methods (GA and GD) result in nearly similar values (except for GA, sk4), whereby those determined with GD are more accurate. These values for GD were achieved after about 33 iterations. The calculation time was approx. 1 hour in most cases.

5 Conclusions

This section shows the final result of the shape optimised according to section 4. Quadrilateral plate elements are used for the deck. Quadrilateral elements are based on the MITC4 formulation published in [6]. The elements allow the calculation of membrane forces and bending moments, i.e. they have a disk and plate effect and are suitable for thin and thick plates. "Phänotyp" also allows the calculation of principal stresses and their direction using these elements. This function, which is also used in additive manufacturing [9], has been specially incorporated to illustrate the load-bearing effect of shell structures for students and users. Figure 11 shows the optimised shape. As an example to illustrate the main stresses, also two other cases with different deck widths are shown. The best structural solution isn’t always the ideal solution from the point of architecture. For use it is important to give quick feedback about the consequences of an architectural shape in the design phase.

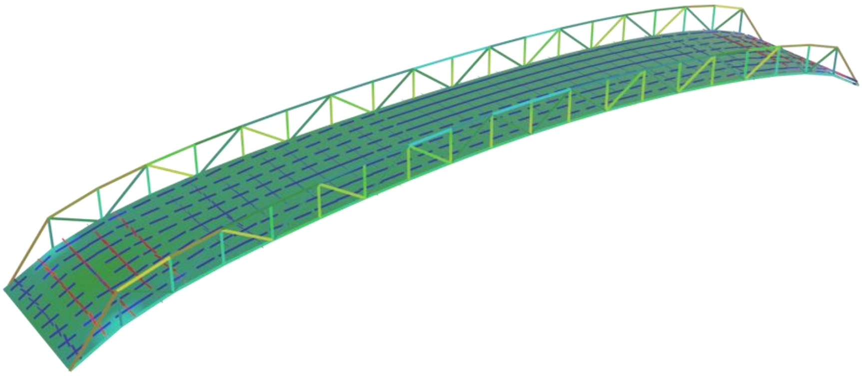

Figure 11: Optimised shape using quadrilateral elements

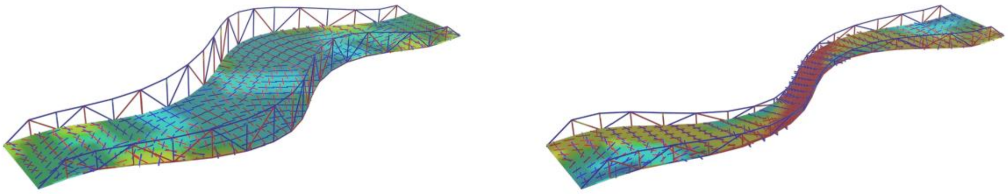

Figure 1. Examples to illustrate stress lines.

References

[1] K. Deix, C. Müller , "A structure topology optimization approach for architects in Blender 3D", in B.H.V. Topping, J. Kruis, (Editors), "Proceedings of the Fourteenth International Conference on Computational Structures Technology", Civil-Comp Press, Edinburgh, UK, Online volume: CCC 3, Paper 5.2,2022,doi:10.4203/ccc.3.5.2, https://www.ctresources.info/ccc/paper.html?id=9403

[2] An 3D structural engineering finite element library for Python, https//github.com/JWock82/PyNite

[3] Blender: https://blender.org

[4] Phänotyp: https://github.com/bewegende-Architektur/Phaenotyp

[5] H. Pohlheim, “Evolutionäre Algorithmen: Verfahren, Operatoren und Hinweise für die Praxis“, Springer, Berlin, 2000

[6] K.-J. Bathe, Finite Element Procedures, 2nd Edition, 4th Printing, Prentice hall, New Jersey, 1996

[7] K-M. Tam, C. T. Mueller, Additive manufacturing along principal stress lines, Massachusetts Institute of rTechnology, DOI: 10.1089/3dp.2017.0001