The Fourteenth International Conference on
Computational Structures Technology
23–25 August, 2022 | Montpellier, France

Proceedings of the Fourteenth International Conference on
Computational Structures Technology
Edited by B.H.V. Topping and J. Kruis
Civil-Comp Conferences, Volume 3, Paper 5.2
Civil-Comp Press, Edinburgh, United Kingdom, 2022, doi: 10.4203/ccc.3.5.2

A structure topology optimization approach for architects in Blender 3D

K. Deix[1], C. Müller[2]

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

Abstract

The presented tool is a shape finding tool for large structures in civil engineering and architecture. Based on a basic structure of nodes and bending members, different shapes can be defined as starting point for the form finding process. Several fitness functions, such as smallest total weight, smallest/uniform longitudinal stress, smallest deformations, large/small lever arm, etc. are provided for shape finding. The resulting multi-dimensional optimization problem is solved using an evolutionary algorithm. This results in novel structural forms that can be used for parametric architectural design [1], among other applications. The tool has a modular structure, allowing different ways to be selected and also to be superposed.

Keywords: generative algorithm, topology optimization, finite element analysis, digital content creation, artistic, architectural, form finding, framework

1 Introduction

The software, called „Phaenotyp", is intended to create a link between early architectural design and structural analysis. Especially in architectural studies it is recognizable that static principles are not yet considered appropriately. But this is where the most influence can be made on efficiency. Instead of a classical CAD program, Phänotyp is integrated in Blender 3D, a program coming from the field of DCC (Digital Content Creation) used in film and game development. These tools are used to think, model and create new worlds, not only in three but four dimensions by including movement. Furthermore we want to address contemporary form finding methods in architecture according to the principles of bionics [2].

"Blender" [3] is a free 3D graphics software. It is used here on the one hand as a GUI (Graphical User Interface), where especially the easy handling and the multiple possibilities to design and modify objects are important. Since Blender is controllable with Python code, the present tool was written in Python and is installable as AddOn [4] in Blender.

The algorithm does not require a discrete design space. The nodes are not fixed in position in space, but move along freely selectable paths (node movement paths). These paths can be freely selected by the functions "rotate", "scale" and "move" of single or multiple nodes (Figure 1). The change of the node positions among each other is additionally possible with selectable functions (e.g. linear, root, smooth, sphere, etc).

The paths are realized with the „Shape-Keys" integrated in Blender, where the node paths, scaled from 0 to 1, can be predefined. By overlaying several shape keys with different paths, unpredictable shapes and a very large number of combinations result.

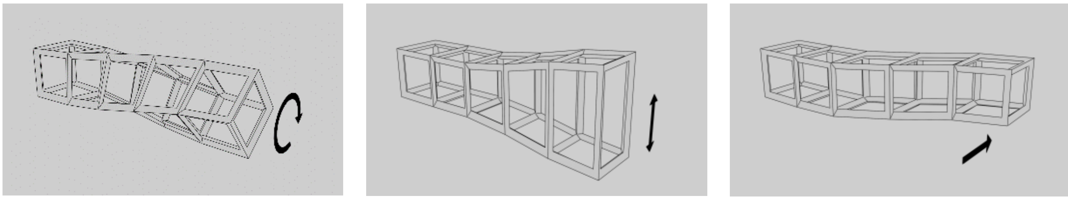

Figure 1: Node movement paths: a rotate, b scale, c move.

Compared to known truss optimization algorithms [5], [6], which are mainly used for additive manufacturing, partly for larger structures [7], the beam calculation and the optimization process are separated. The framework calculation is performed by Finite Element Analysis using the Python program "Pynite" [8], which is also freely available. The calculated structure consists of frames, which are composed of 3d beams. Flexural beams are widely used in civil engineering and are important for architectural applications. In contrast to truss structures, which have only axial loads, whose nodes are jointed, and which must always be composed of triangles, the shape of the structure in flexural beams is unrestrained. Furthermore, not only nodal loads but also distributed loads are possible and several optimization options are available. The disadvantage, on the other hand, is the longer calculation time, since a member has 12 instead of 3 degrees of freedom and thus the stiffness matrix becomes significantly larger.

2 Methods

The basic model created in Blender is assigned with profiles, loads, forces and supports. The result of finite element analysis using Pynite are the axial, shear, bending and torsional moments as well as displacement at the end nodes i of the beam (N_i, V_{yi}, V_{zi}, M_{yi}, M_{zi}, M_{it}, w_i). From this, the longitudinal and shear stresses and their superposition are calculated at 10 (j from 0 to 10) sections of each member, and the utilization ratio α_i is determined using the linear elastic limit states specified for the selected materials and the allowed stress. This is exemplified in formulas 1 and 2 for the case of longitudinal normal stresses:

$$\sigma_{i,j} = \pm \frac{N_i}{A_i} \pm \frac{M_{i,j}}{W_i} \quad (1)$$

$$\alpha_i = max\left(\frac{\sigma_{i,j}}{\sigma_{allow}}\right) \quad (2)$$

Furthermore, Euler buckling under compressive loading is taken into account by reducing the allowable stress as a function of slenderness. The shear forces due to shear force and torsion as well as the Van- Mises equivalent stress are also calculated.

The following modules are currently available:

- Single Frame Analyse

Simple finite element analysis with numerical output and graphical representation of the above characteristic values.

- Animation

Enables the analysis of a moving structure. Thus, worst case situations of kinematic structures can be found.

- Sectional Performance of the cross-section with a tanh-function as following:

$$f_i = 1 + 0{,}36 \cdot \alpha_i \qquad \alpha_i > 1$$
$$f_i = 0.5 + 0{,}6 \cdot (tanh(\alpha_i - 0.5) \cdot 2.4) \qquad 0 \leq \alpha_i \leq 1 \quad (3)$$

As a result, lightly loaded members are assigned smaller cross-sections and other members receive a more uniform utilization.

- Shape-Optimization

The shape optimization is done with the above mentioned node movements paths. Several fitness functions are implemented, which can be minimized.

Biggest or average stress in a beam (to optimize the cross sections):

$$\max(\sigma_i)\,,\ \sum_{i=0}^{n} \frac{\max(\sigma_i)}{l_i} \qquad (4)$$

Lowest total mass (for material savings):

$$\sum_{i=0}^{n} A_i \cdot l_i \cdot \rho \quad (5)$$

Average lever arm (large lever arms lead to bending beams with large cross-sectional heights (such as I-beams), small lever arms lead to compact cross-sections):

$$\sum_{i=0}^{n} \frac{\max(M_i)}{N_i \cdot n} \quad (6)$$

Planned maximum deflection:

$$\max(w_i) \quad (7)$$

- Genetic Algorithm

The multidimensional optimization process uses a genetic algorithm [9], [10]. The idea is that the pairing of existing shapes can lead to new solutions just like in evolution. In this algorithm, the value of a shape key functions as seen as a gene. A combination of several shape keys as chromosomes. A population is a pool of individuals with different chromosomes. Based on the given fitness function, the individual chromosomes are sorted according to their performance. Elitism means that the best percent are directly transferred to the next population. The other individuals are mated and their children are transferred to the next population (fig.2).

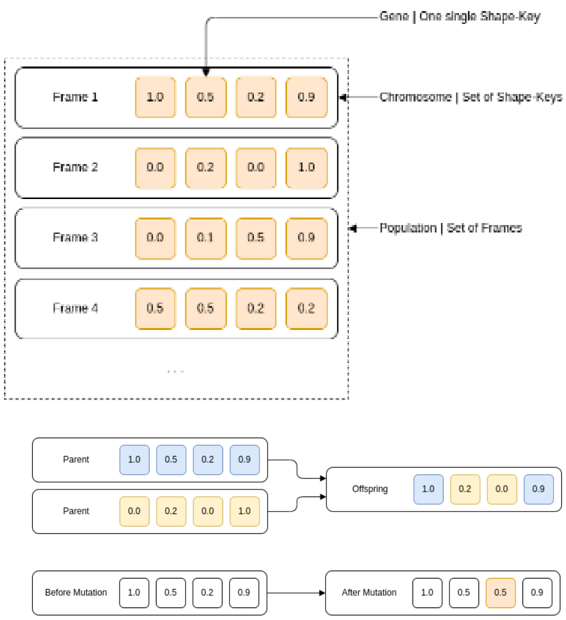

Figure 2: Genetic algorithm

3 Results

Phänotyp is currently also able to optimize the topology of a structure. The Decimate modifier available in Blender-3D is used for this purpose. As a basis for the decimation of edges, a vertex group is created based on a previous calculation. The first image shows a simple structure that was analyzed and reduced using this method.

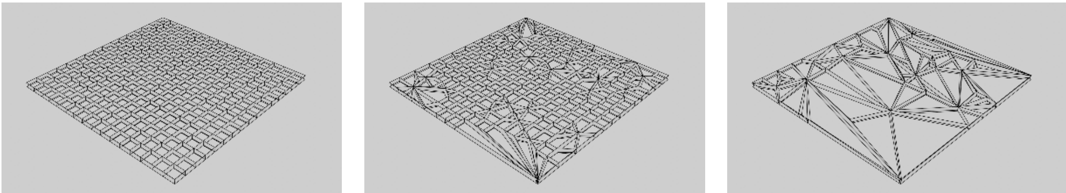

Figure 3: Topology-Optimization through decimation.

The second example shows a more complex shape created using this method. Here, optimisation by gradient descent could still be applied.

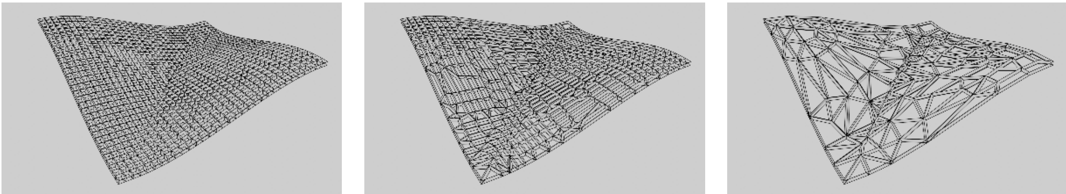

Figure 4: Topology-Optimization through decimation.

The algorithm is illustrated by a simple example. A beam with frame-like beams with supports at the ends (Figure 5) is given. In addition to the self-weight, vertical point loads at the upper nodes and a single point load in the middle in the horizontal direction were selected as loads. The three selected shape keys (Fig 5) allow an arc-like increase, an increase of the cross-section height in the center of the field and a rotation in the middle of the beam.

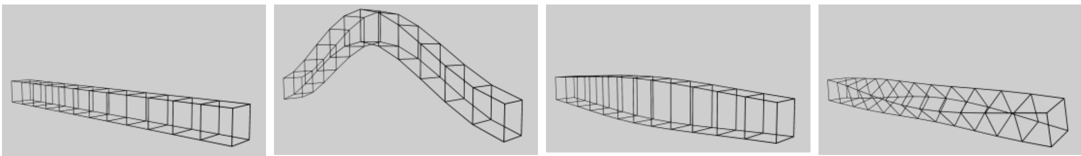

Figure 5: Pre defined shape-keys.

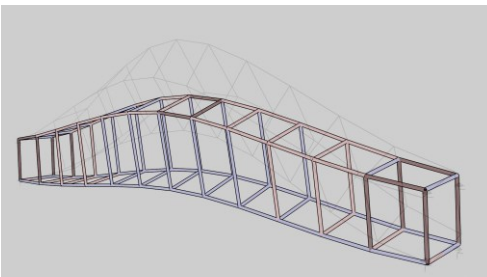

Figure 6: Result of the multidimensional optimization

Figure 6 shows the result of the multidimensional optimization process with three shape keys. It can be seen that a small arch-like curvature of the shape key 1 (= 0.2) is favorable, also a transverse section increase with shape key 2 (= 1.0) in the center of the beam. The rotation of the beam (shape Key 3= 0.5) is due to the applied horizontal concentrated load. In the right image of Figure 7, several iterations were subsequently performed with the section optimizer. Figure 8 shows the stresses of some members during the first iterations.

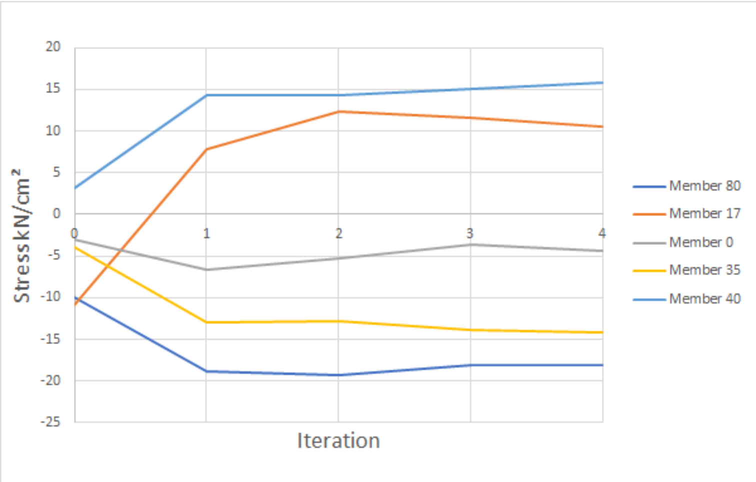

Figure 7: Stresses of chosen members during the first iterations.

4 Conclusions and Contributions

In this thesis, we presented a topology optimization tool for shape finding for large structures in civil engineering and architecture. Based on a basic structure of nodes and bending members, different shapes can be defined, which can also overlap. Several fitness functions, such as smallest total weight, smallest/uniform longitudinal stress, smallest deformations, large/small lever arm, etc. are provided for shape finding. The resulting multi-dimensional optimization problem is solved using an evolutionary algorithm. This results in novel structural forms that can be used for parametric architectural design, among other applications. The tool has a modular structure, allowing different ways to be selected and also to be superposed. The tool is intended to create a link between early architectural design and structural analysis [11].

References

[1] P. Schumacher, “Parametricism - A New Global Style for Architecture and Urban Design”, AD Architectural Design - Digital Cities, Vol 79, No 4, London, 2009.

[2] M. Colletti, B. Massin, “meeting nature halfway: architecture interfaced between technology and environment”, innsbruck university press, 2018.

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

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

[5] C. Reintjes, “Algorithm-Driven Truss topology Optimization for Additive manufacturing”, Springer Vieweg, Wiesbaden, 2021
[6] C. Lemaitre, “Topologieoptimierung von adaptiven Stabwerken, Dissertation“, Universität Stuttgart, 2008
[7] E. Ching, J.V. Carstenesen, “Truss topology optimization of timber-steel structures for reduced embodid carbon design”, Engineering Structures, 252, 2022, https://doi.org/10.1016/j.engstruct.2021.113540
[8] A 3D structural engineering finite element library for Python, https://github.com/JWock82/PyNite
[9] A.Kumar, “Genetic Algorithms“, https://www.geeksforgeeks.org/genetic-algorithms/
[10] H. Pohlheim, “Evolutionäre Algorithmen: Verfahren, Operatoren und Hinweise für die Praxis“, Springer, Berlin, 2000
[11] M. Carpo, “The Alphabet and the Algorithm”, The MIT Press, 2011