Article:
He, L. orcid.org/0000-0002-2537-2244, Fairclough, H.E. orcid.org/0000-0003-3133-1534, Gilbert, M. orcid.org/0000-0003-4633-2839 et al. (2 more authors) (2025) Minimum material vault designs generated via adaptive layout optimization. Engineering Structures, 334. 120213. ISSN 0141-0296

https://doi.org/10.1016/j.engstruct.2025.120213

Minimum material vault designs generated via adaptive layout optimization

Linwei He[a], Helen E. Fairclough[a], Matthew Gilbert[a], Andrew Liew[b], Karol Bołbotowski[c,d]

[a] *School of Mechanical, Aerospace and Civil Engineering, The University of Sheffield, Mappin Street, Sheffield, S1 3JD, UK*
[b] *Unipart Construction Technologies Ltd, Advanced Manufacturing Park, Sheffield, S60 5WG, UK*
[c] *Department of Structural Mechanics and Computer Aided Engineering, Faculty of Civil Engineering, Warsaw University of Technology, 16 Armii Ludowej Street, Warsaw, 00-637, Poland*
[d] *Lagrange Mathematics and Computing Research Center, 103 rue de Grenelle, Paris, 75007, France*

Abstract

Vaults are commonly used to form lightweight long-span roof structures, allowing flexible internal spaces with minimal associated embodied carbon. The precise shape of the vault should be chosen to reduce or eliminate bending effects, so as to promote more-efficient structures that work in pure compression. Many existing form-finding methods can identify bending-free designs; however, these are restricted to operate on predefined layouts and therefore cannot generally achieve optimal material-efficiency. This paper presents a new family of form-finding methods employing the numerical layout optimization method, which uses the 'ground structure' approach to simultaneously optimize a vault's form and force flow topology. A conic programming problem is formulated, enabling the attainment of globally-optimal minimum-volume designs for any given nodal discretization. The formulation is newly presented from an accessible engineering perspective, building upon the standard truss layout optimization method to provide an explainable and flexible framework for future research within the community. To enhance computational efficiency, an adaptive 'member adding' technique is employed, enabling the solution of large-scale problems while also allowing rapid exploration of smaller-scale scenarios for more practical vault designs. The proposed method is applied to a range of examples, demonstrating the ability of the proposed procedure to generate more materially efficient vault designs, compared to traditional Force Density Method (FDM) designs.

Keywords: form-finding, layout optimization, vaults, truss topology optimization, ground structure method

1. Introduction

Vaults are three-dimensional arch structures, typically constructed in masonry, and commonly used to form lightweight, long-spanning roof structures and ceilings. Their structural efficiency stems from their geometric form that uses building materials effectively, and is balanced with providing aesthetically pleasing architectural designs. Careful consideration of the form of a vault, can eliminate bending stresses and promote a compression only internal stress state, thereby improving structural efficiency and eliminating the need for tensile reinforcement in its design. To aid in the design of arched and vaulted geometries, various physical and numerical form-finding methods have developed in the past. Early works in form-finding can date back to the 6th century AD [1] and gained widespread recognition through physical-based hanging chain (Antonio Gaudí), soap film (Frei Otto) and hanging cloth (Heinz Isler) models in designing funicular structures carrying only compressive forces. Numerical form-finding methods were later developed, for which the classical approaches can be categorized into three main families (Adriaenssens et al. [2], Veenendaal and Block [3]),

- Stiffness matrix methods (Sive and Eidelman [4]),
- Geometric stiffness methods such as the Force Density Method (FDM) (Schek and Linkwitz [5][6][7]) and Thrust Network Analysis (TNA) (Block and Ochsendorf [8], Block and Lachauer [9], O'Dwyer [10], Fraternali [11], Marmo and Rosati [12]),
- Dynamic equilibrium methods, including Dynamic Relaxation (DR) (Barnes [13], [14], [15]) and particle-spring systems (Kilian and Ochsendorf [16]).

However, these traditional form-finding methods require the designer to pre-define the structural layout at the start of the process, which in the case of vaults, is an assumption on the compressive load paths or force flow in the structure. This means that the designs generated are unlikely to constitute minimum material solutions, based instead on historical best practise or past experience of "good" solutions. Additionally, performing form-finding with such a given structural topology, methods such as the FDM and DR will move the nodes in all spatial directions to maintain equilibrium with the externally applied loads, leading to a final form-found shape that does not maintain its initial plan projection. This can result in form-found structures that are inconsistent with their starting assumptions on self-weight loading and make it difficult to isolate only vertical movements in the form-finding of the vault structure. TNA resolves this issue by separating the force system's planar equilibrium from the elevation form-finding, and can additionally be augmented with minimizing material volume [17, 18], optimizing stability [19], or assessing structural safety [20], by optimizing force densities. If a FDM approach is adopted for optimizing force densities for minimum volume, constraints must be applied to control the plan projected movements with a Constrained FDM (CFDM) [21]. However, there are limitations to these methods of minimizing structural volume with TNA or FDM, namely the use of slower non-linear optimization solvers that have difficulty with scaling for larger problems, and still the inability to optimize the topology of the force system itself, relying on a pre-defined layout. These limitations can be partially addressed by iteratively alternating network connectivities and force densities [22, 23], though this approach relies on trial and error, and the number of pre-defined layouts still remains limited, depending on the designer's experience.

To remove this dependence on predefined layouts and member force densities in a compressive only truss network, here a new family of form-finding methods, capable of identifying optimal structural layouts in parallel with the form-finding processes, is sought. In this contribution the layout optimization method is employed. Since being developed in the 1960s for truss structures [24], the layout optimization methods have been applied to various structural systems, including building bracings [25], long span bridges [26], and floor grillages [27]. Furthermore, an adaptive 'member adding' scheme was invented by Gilbert and Tyas [28], which allowed solutions to be obtained more quickly, and also for extremely large problems (e.g., over one billion members) to be solved. This permits layout optimization to be used as a numerical means of identifying theoretical minimum material designs of structures.

However, whereas in a conventional topology optimization problem the location of the applied loading is known *a priori*, as mentioned previously, for form-finding problems this is not the case, as the structural form and its resulting self-weight loading has not been defined. To address this, Fuchs and Moses [29] proposed a 'transmissible loads' formulation, in which the vertical position of a given load could be varied (albeit issues with this formulation were later highlighted [25]). Layout optimization has previously been applied to form-finding problems for arch structures using the aforementioned 'transmissible loads' [30]. However, the solutions obtained may comprise of multiple surfaces or layers in the resulting structure; furthermore, the use of 3D design spaces significantly increases computational cost. Jiang [31] introduced a reduced ground structure with an aim to improving computational efficiency, though the mathematical rigour is also compromised due to the heuristics involved in shrinking the 3D design domain and the presence of tensile regions.

Recently, a theoretical and computational breakthrough in form-finding was made in Bołbotowski [32]. At the conceptual level, the work was inspired by the optimal archgrid theory put forth by Rozvany [33], see also more recent developments [34, 35]. Similar to some of the aforementioned methods, the archgrid approach requires a pre-fixed layout. These restrictions were removed in [32], which provides a rigorous theory of 3D vaults that globally minimize the material volume. In general, these vaults are mixtures of framed grid-shells and membrane shells, being reminiscent of the renown Michell structures [36, 37], also hybrid in nature. The key idea behind the contribution [32] consists in rewriting the 3D form-finding to a 2D convex problem. The latter formulation serves as a springboard to powerful numerical methods: well posed, due to convexity, and fast, owing to the reduced dimension of the design space. The paper capitalizes on that by employing the layout optimization via a ground structure. It leads to a second order conic programming (SOCP) problem as opposed to classical ground structure formulations [24]-[28] being linear programs. Nevertheless, it facilitates a similar level of efficiency, especially when boosted with the adaptive member adding scheme, also implemented and demonstrated in [32]. Eventually, a solution of the 2D layout optimization problem furnishes a highly precise grid-shell approximation of the theoretically optimal vault.

Although the underlying optimization formulation was presented in [32], its engineering interpretation was potentially overshadowed by the presence of numerous mathematical proofs. Consequently, researchers without a strong mathematical background may find it challenging to fully comprehend and adapt the method in future work. Therefore, the goal of this paper is to make this new family of form-finding methods significantly more accessible to the community by offering a new perspective, deriving the optimization problem through accessible engineering concepts. Firstly, the exposition of the numerical method will be greatly simplified. In [32] the SOCP formulation was put forth as a way of approximating an auxiliary abstract formulation. In contrast, this paper derives the problem starting from classical truss layout optimization, with a simple example to elucidate the engineering characteristics of the problem.

Secondly, the applicability of the method is demonstrated through a benchmark-to-practical design workflow. While the full 'ground structure' is employed to obtain benchmark designs, more practical designs are generated by restricting the connectivity of the initial ground structure. This simple modification allows generating more practical designs with only slight increase of the material volume. The efficacy of the new form-finding method is then demonstrated through comparative studies against the FDM, with uniform or optimized force density distributions, where in the examined examples the herein presented method will produce designs of significantly lower volume and with reduced computational cost. Thirdly, the practicality of the method is further improved here through the addition of various support conditions, such as symmetry and roller supports, as well as load conditions, including point loads and patch loads. These simple modifications illustrate that the underlying formulation can be readily adapted for future research. Finally, the paper comes with a Python script in which the form-finding method is implemented for the reader to explore.

The paper is organised as follows. In Section 2, the vault layout optimization procedure is outlined and the adaptive member adding process explained. In Section 3, various numerical examples are solved to verify the new method, and to compare it with classical force density methods. Finally, conclusions are drawn in Section 4.

2. Methods

The standard truss layout optimization is introduced first and then used as a basis for deriving the SOCP formulation for vaults.

2.1. The standard truss layout optimization

The standard layout optimization process involves four steps, as shown in Figure 1. Firstly, the design domain, loading and support conditions are specified, Figure 1(a); secondly, the design domain is discretized using a general or regular grid of nodes, Figure 1(b); thirdly, a 'ground structure' is created by interconnecting nodes with potential truss members, Figure 1(c); finally, an optimum layout is identified by solving the following optimization problem:

$$\min_{\mathbf{q},\mathbf{a}} \quad V = \mathbf{l}^{\mathrm{T}}\mathbf{a}, \tag{1a}$$
$$\text{s.t.} \quad \mathbf{B}\mathbf{q} = \mathbf{f}, \tag{1b}$$
$$-\sigma\mathbf{a} \le \mathbf{q} \le \sigma\mathbf{a}, \tag{1c}$$

where, V is the structural volume, $\mathbf{a} = [a_1, a_2, ..., a_m]^{\mathrm{T}}$ is a vector containing member cross-sectional areas, with m denoting the number of members. $\mathbf{l} = [l_1, l_2, ..., l_m]^{\mathrm{T}}$ is a vector of member lengths, $\mathbf{B}$ is a $2n \times m$ equilibrium matrix comprising direction cosines, with n denoting the number of nodes and $\mathbf{q}$ is a vector containing the internal member forces. Finally $\mathbf{f}$ is a vector containing the external forces with σ as the limiting material stress. Problem (1) is a linear programming (LP) problem with respect to state variable $\mathbf{q}$ and design variable $\mathbf{a}$. Note that Problem (1) does not adjust the coordinates of nodes; thus, the resulting structural form is influenced by the nodal grid defined in the 'ground structure'. Nevertheless, due to the vast number of potential layouts within the 'ground structure', even a coarse nodal grid can lead to highly accurate solutions, as demonstrated in previous studies (e.g., [38]). When a finer nodal grid is employed, the solution converges to the theoretical minimum, as illustrated in works such as [30].

2.2. Adaptation to form-finding

2.2.1. Compression only structure

Having in mind that compression is favoured in vaults, each compressive force q is desired to be positive. If a structure is in pure compression, then the cross-section area of each element can be chosen to ensure that the limiting stress is reached, i.e. $\mathbf{a} = \frac{\mathbf{q}}{\sigma}$. This can be used to eliminate design variable $\mathbf{a}$:

$$\min_{\mathbf{q}} \quad V = \frac{1}{\sigma}\mathbf{l}^{\mathrm{T}}\mathbf{q}, \tag{2a}$$
$$\text{s.t.} \quad \mathbf{B}\mathbf{q} = \mathbf{f}, \tag{2b}$$
$$\mathbf{q} \ge \mathbf{0}. \tag{2c}$$

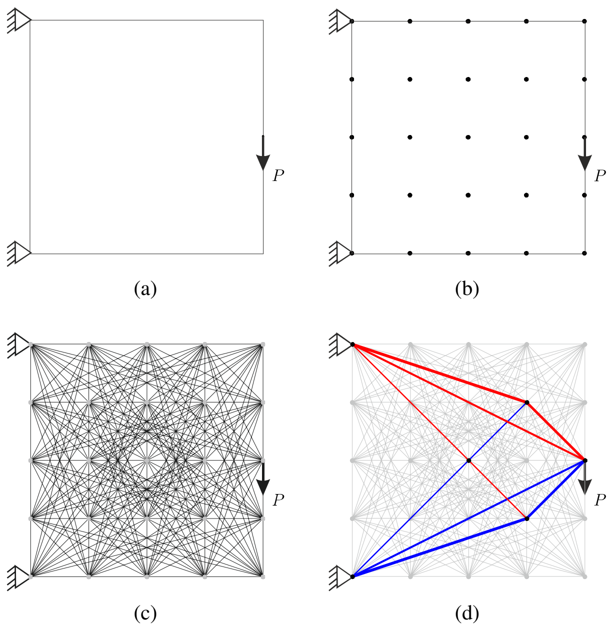


Figure 1: Steps in 2D truss layout optimization.

2.2.2. Node elevation

A vault will be assumed to be a vertically loaded 3D truss that is erected from a planar horizontal ground structure. The structure will be supported at the nodes whose position is fixed within the starting base plane. The remaining free nodes may be moved vertically, and their elevations are then considered as extra design variables.

The nodal elevations will be another variable vector $\mathbf{z} = [z_1, z_2, ..., z_n]^{\mathrm{T}}$. The difficulty with respect to the standard ground structure formulation is that now, with the nodes vertically moving, the length vector $\mathbf{l}$ and the equilibrium matrix $\mathbf{B}$ are not constants as they depend on $\mathbf{z}$. The form-finding formulation thus reads:

$$\min_{\mathbf{q},\mathbf{z}} \quad V = \frac{1}{\sigma}\mathbf{l}^{\mathrm{T}}(\mathbf{z})\mathbf{q}, \tag{3a}$$
$$\text{s.t.} \quad \mathbf{B}(\mathbf{z})\mathbf{q} = \mathbf{f}, \tag{3b}$$
$$\mathbf{q} \ge \mathbf{0}, \tag{3c}$$

where the load vector has the structure $\mathbf{f} = [0, 0, f_{1,z}, ..., 0, 0, f_{n,z}]^{\mathrm{T}}$.

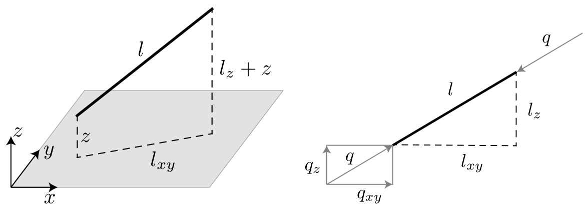
Formulation (3) is not computationally efficient to solve with optimization solvers, as the extra variable $\mathbf{z}$ renders the problem non-convex, therefore ruling out employing efficient solvers that are known to work very well for large-scale standard ground structure settings. Accordingly, further reformulation is required.

2.3. SOCP reformulation

As shown in Figure 2, member length can be calculated via,

$$l = \sqrt{l_{xy}^2 + l_z^2} = l_{xy}\sqrt{1 + \frac{l_z^2}{l_{xy}^2}} = l_{xy}\sqrt{1 + \frac{q_z^2}{q_{xy}^2}}, \tag{4}$$

where l_{xy} is the projected member length on xy plane; l_z represents the difference in nodal elevations between the two end



(a) (b)

Figure 2: An elevated member and its internal forces

nodes; q_{xy} and q_z are the revolved member forces in xy plane and z direction, respectively. Note that the final representation in terms of q is possible since the purely axial loading ensures that the inclination of the axial force must be identical to the inclination of the bar, i.e. $\frac{l_z}{l_{xy}} = \frac{q_z}{q_{xy}}$.

Since the structure is in pure compression, i.e., $q_{xy} \geq 0$, the internal force of a member can be written as,

$$q = \sqrt{q_{xy}^2 + q_z^2} = q_{xy}\sqrt{1 + \frac{q_z^2}{q_{xy}^2}}. \tag{5}$$

Therefore, by substituting equations (4) and (5) into the objective function (2a), the volume of a single member can be calculated using,

$$v = \frac{lq}{\sigma} = \frac{l_{xy}q_{xy}}{\sigma}\left(1 + \frac{q_z^2}{q_{xy}^2}\right) = \frac{l_{xy}}{\sigma}\left(q_{xy} + \frac{q_z^2}{q_{xy}}\right) \tag{6}$$

where v is the member volume. Introduce now an auxiliary variable r and let,

$$r \geq \frac{q_z^2}{2q_{xy}}, \tag{7}$$

For an optimal solution, the equality condition in (7) must hold, then equation (6) can be written as,

$$v = \frac{l_{xy}}{\sigma}\left(q_{xy} + 2r\right), \tag{8}$$

which is now a linear expression with respect to variables q_{xy} and r, and the elevations $\mathbf{z}$ are no longer present.

On the other hand, inequality constraint (7) can be reformulated as

$$2rq_{xy} \geq q_z^2, \tag{9}$$

which is a rotated quadratic conic constraint that can be solved efficiently via SOCP.

Just as the elevation vector $\mathbf{z}$ was eliminated from the objective function (2a), utilizing the component vectors $\mathbf{q}_{xy}, \mathbf{q}_z$ instead of $\mathbf{q}$ splits the equilibrium equation (2b) into two linear equations. Ultimately, it leads to the SOCP formulation for form-finding via layout optimization, which is significantly less challenging to solve than the initial problem (3),

$$\min_{q_{xy},q_z,r} \quad V = \frac{1}{\sigma}\mathbf{l}_{xy}^{\mathrm{T}}\left(\mathbf{q}_{xy} + 2\mathbf{r}\right), \tag{10a}$$

$$\text{s.t.} \quad \mathbf{B}_{xy}\mathbf{q}_{xy} = \mathbf{0}, \tag{10b}$$

$$\mathbf{B}_z\mathbf{q}_z = \mathbf{f}_z, \tag{10c}$$

$$2rq_{xy} \geq q_z^2 \quad \text{for all members}, \tag{10d}$$

where, $\mathbf{l}_{xy}$, $\mathbf{q}_{xy}$, $\mathbf{q}_z$ and $\mathbf{r}$ are $m \times 1$ vectors containing the corresponding variables of all members. $\mathbf{f}_z$ is a vector of vertical point loads applied at nodes. Additional horizontal point loads may also be applied to the force equilibrium constraint (10b), but in this context, they are considered in the same load case as $\mathbf{f}_z$, thus limiting their usefulness in practical scenarios. For uniformly distributed loads (UDL), the equivalent lumped point loads should be applied. If the UDL is intended to approximate self-weight, the lumped point loads may not always provide accurate results, and alternative approaches are required, which are the subject of ongoing research. $\mathbf{B}_{xy}$ is a $2n \times m$ equilibrium matrix on the xy plane and $\mathbf{B}_z$ is a $n \times m$ equilibrium matrix in the z axis. The local equilibrium matrices for a member i can be written as,

$$\mathbf{B}_{xy,i} = \begin{bmatrix} -\Delta x/l_{xy} \\ -\Delta y/l_{xy} \\ \Delta x/l_{xy} \\ \Delta y/l_{xy} \end{bmatrix}, \tag{11}$$

and

$$\mathbf{B}_{z,i} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \tag{12}$$

where Δx and Δy denote the differences in the x and y coordinates of the end nodes of member i.

It is worth mentioning that the SOCP formulation (10), primarily derived from the plastic minimum volume formulation, is equivalent to the elastic design of vaults, i.e. to a minimum compliance problem. This was evidenced in Section 6.2.3 of [32]. In particular, there exist admissible nodal displacements that generate a uniform axial strain in all the members, i.e. the elastic structure is uniformly stressed.

2.4. *A very simple example*

To provide some intuition on the characteristics of the problem (10), a simple example will be considered. This example is as shown in Fig. 3. The ground structure here contains just 2 elements, both of which are aligned to the x-axis. The allowable stress σ for this example is 1. Thus, for this case, problem (10) becomes:

$$\min_{q_{xy},q_z,r} \quad V = \begin{bmatrix} 2 \\ 1 \end{bmatrix}^{\mathrm{T}}\left(\begin{bmatrix} q_{xy,1} \\ q_{xy,2} \end{bmatrix} + 2\begin{bmatrix} r_1 \\ r_2 \end{bmatrix}\right), \tag{13a}$$

$$\text{s.t.} \quad \begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} q_{xy,1} \\ q_{xy,2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{13b}$$

$$\begin{bmatrix} 1 & -1 \end{bmatrix}\begin{bmatrix} q_{z,1} \\ q_{z,2} \end{bmatrix} = \begin{bmatrix} 1 \end{bmatrix}, \tag{13c}$$

$$2r_1q_{xy,1} \geq q_{z,1}^2 \quad , \tag{13d}$$

$$2r_2q_{xy,2} \geq q_{z,2}^2 \quad , \tag{13e}$$

where additional subscripts indicate whether the variable relates to element 1 or 2.

Firstly, note that the equilibrium conditions are written only for the unsupported degrees of freedom at point B. At points A

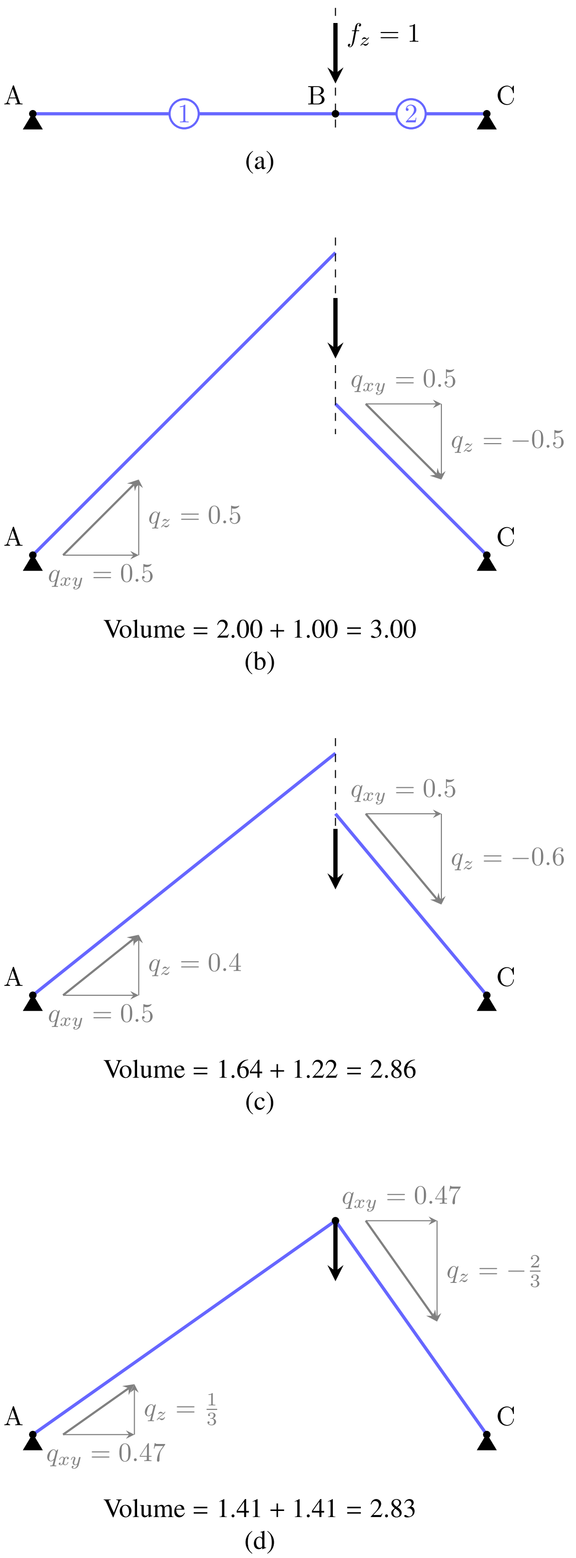


Figure 3: Very simple example: (a) problem specification; (b) a non-optimal but feasible (i.e., satisfying all constraints in (13)) solution; (c) another non-optimal but feasible solution; (d) the optimal solution. Volumes of each element and total volume shown.

and C, the freely chosen reaction forces mean that equilibrium will always be satisfied there, and thus those constraint rows should be removed from the problem. The in-plane equilibrium constraint (13b) has two rows, corresponding to the x and y direction at point B; but as there are no elements in this example which have a component in the y direction, the second row is trivially satisfied as 0 = 0.

Suppose now that a feasible solution to (13) is sought. As an initial guess, it is assumed that element 1 will carry half the vertical load, $q_{z,1} = 0.5$. Vertical equilibrium constraint (13c) then implies that $q_{z,2} = -0.5$, i.e. element 2 carries the other half of the load, with a downwards slope. The horizontal thrust is initially guessed to be 0.5, with the in-plane equilibrium constraint (13d) ensuring $q_{xy,1} = q_{xy,2}$.

However, if we attempt to draw this solution, as has been done in Figure 3b, it quickly becomes apparent that this does not form a valid structure. Both elements have vertical and in-plane forces of equal magnitude, and so to keep these in line with the elements, they must both have a 1:1 slope (upwards or downwards respectively). This means that they arrive at point B at different heights. Conceptually, it should be imagined that a cost-free, infinitely stiff vertical bar joins the elements here (shown as dashed in Fig. 3(b)). Clearly this is not a practical representation of the real-world, however it will be shown that this issue vanishes as we improve the optimality of the solution.

An intuitive improvement on the initial solution above/in Fig. 3(b) would be to increase the proportion of loading transmitted to the closer support. Fig. 3(c) shows the structure obtained by setting $q_{z,1} = 0.4$. The volume of element 1 has been reduced by 0.36 while the volume of element 2 increases by just 0.22, leading to an overall reduction in the objective function. Furthermore, by drawing the elements aligned to their respective axial force vectors, it can be seen that the vertical discontinuity at B has reduced.

The optimal solution to (13) is shown in Fig. 3(d). For the optimal solution there is no discontinuity at point B, i.e. the values of q_{xy} and q_z represent a true single-layer structure. A key theoretical breakthrough in [32] was to prove that this is the case for all problems, also when the ground structure is truly two dimensional. This is discussed in the next subsection. Figure 4 shows the main steps in layout optimization of vaults employing a fully connected ground structure; both the optimum member layout and the elevations of nodes are obtained.

2.5. *Recovering the elevations of the nodes*

The nodal elevations $\mathbf{z}$ are absent from the SOCP formulation (10). Nonetheless, after finding the solution $\mathbf{q}_{xy}$, $\mathbf{q}_z$, $\mathbf{r}$ the elevations must be restored in order that the final 3D design is generated. From Figure 2 follows that, for every bar in the ground structure that is of non-zero member force, the ratio $\frac{q_z}{q_{xy}}$ determines the slope of the bar. More precisely,

$$l_z = l_{xy}\frac{q_z}{q_{xy}} \qquad \text{if} \quad q_{xy} > 0. \tag{14}$$

For each member l_z is simply a difference between the end elevations. It is possible to obtain elevations of nodes by following the slope of members, starting from fixed nodes. However,

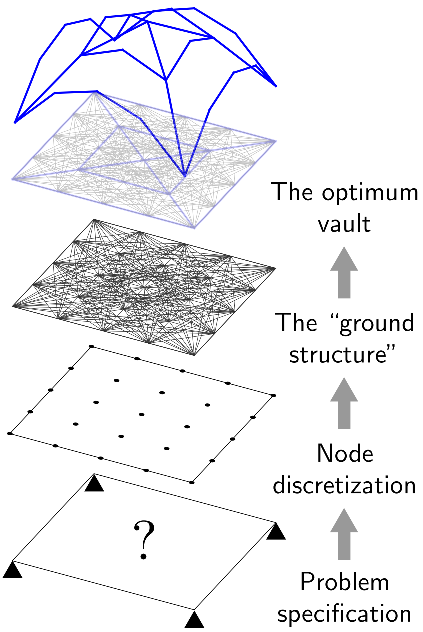


Figure 4: Layout optimization of vaults employing a ground structure method.

note that this equation itself does not guarantee that the outcome vault is single-layered. It is possible that members meet at different elevations that a node is split into multiple nodes in the z direction, for example see Figure 3(b). To address this, it can be shown that there holds the linear geometric relation,

$$\mathbf{B}_z^{\mathrm{T}}\mathbf{z} = \mathbf{l}_z. \tag{15}$$

Combining (14) and (15) leads to a linear system for the unknown elevations $\mathbf{z}$. Since $m \gg n$ for fully connected ground structure, this system is largely overdetermined. Thus, for the system to be consistent, the vector of ratios $\frac{q_z}{q_{xy}}$ must satisfy certain compatibility conditions, which are not a priori built into the problem (10). It is important to note that a significant theoretical breakthrough in [32] is that the study proves that the consistency of the linear system is guaranteed whenever $\mathbf{q}_{xy}$, $\mathbf{q}_z$ are optimal for (10) (Theorem 6.6 in [32], see also the simple example in Figure 3). In [32] this was shown via the dual solution (cf. Appendix B) which also automatically delivers the compatible elevations $\mathbf{z}$. In this paper a direct analysis of the problem (10) is performed in Appendix A to show the well posedness of the elevations' recovery problem.

Since it is proved that a consistent linear system (15) is guaranteed when the optimum solution of (10) is found, various approaches can be developed to identify the elevations of nodes. For example, it is possible to use the slope of members in (14) to obtain the elevation of nodes from one to another. Alternatively, the least-square solution of (15) can be used, which is implemented in the supplemented Python script. Note that least-square solution only provides a simple means of solving system (15); as discussed above, since the system is consistent, the residual error is negligible.

For the very simple example in Sect. 2.4, the system to be solved can be written as:

$$\begin{bmatrix} 1 \\ -1 \end{bmatrix} \begin{bmatrix} z_B \end{bmatrix} = \begin{bmatrix} 2\frac{q_{z,1}}{q_{xy,1}} \\ 1\frac{q_{z,2}}{q_{xy,2}} \end{bmatrix}. \tag{16}$$

For the optimal solution in Fig. 3d, this system is consistent, allowing for $z_B = 1.41$:

$$\begin{bmatrix} 1 \\ -1 \end{bmatrix} \begin{bmatrix} z_B \end{bmatrix} = \begin{bmatrix} 2\frac{0.33}{0.47} \\ 1\frac{-0.67}{0.47} \end{bmatrix} = \begin{bmatrix} 1.41 \\ -1.41 \end{bmatrix}. \tag{17}$$

However, for the non-optimal solutions the system cannot be solved. E.g. for the solution in Fig 3(b) it becomes

$$\begin{bmatrix} 1 \\ -1 \end{bmatrix} \begin{bmatrix} z_B \end{bmatrix} = \begin{bmatrix} 2\frac{0.5}{0.5} \\ 1\frac{-0.5}{0.5} \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \end{bmatrix}. \tag{18}$$

Also note that [32] provides a simpler method that directly identifies the elevation of nodes without the need of solving (15), though some more complex derivations are involved. For sake of simplicity it is not discussed here.

2.6. *Adaptive 'member adding' scheme*

Adaptive 'member adding' [28] is a technique used in various layout optimization methods to significantly improve computational efficiency without losing optimality (e.g., [27, 30, 38]). Instead of solving a problem with all members in the 'ground structure' (e.g., Figure 1 (c)), the member adding scheme first 'activates' only a small subset of members (i.e., members used to solve problem (10)). The solution is then improved by iteratively activating new members, which are identified by examining the dual problem of (10), see Section Appendix B.2 for details. A flowchart of the 'member adding' scheme is illustrated in Figure 5. Here the starting member subset contains members with a length up to $\sqrt{2}$ times the nodal spacing, i.e. members parallel or at 45° to the x and y directions. Note that while the choice of the initial subset is nonunique, the solution process and total computational cost will remain similar. It is even possible to start from an infeasible (i.e., force equilibrium cannot be satisfied) subset, though some extra steps are required; see Appendix B.3 for details. The iterative process terminates when the dual problem cannot identify any new members to further improve the solution. It is important to note that, the use of dual problem ensures that the solution obtained via the 'member adding' scheme is equivalent to that derived via the full problem, and the choice of initial subset has no impact on the final solution. An example is given in the following section.

3. Numerical examples

To demonstrate the efficacy of the proposed layout optimization method, several numerical examples are solved. The volume results are normalized using the magnitude p of the UDL, problem dimension L and limiting stress σ. All quoted CPU costs are obtained using a workstation equipped with Intel Xeon E5-2680v2 processors running 64-bit CENTOS Linux.

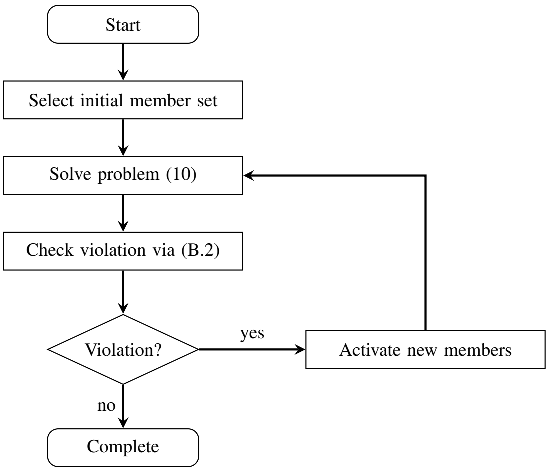


Figure 5: Flowchart of the 'member adding' scheme.

3.1. Verification example: square domain with corner supports

To verify the method, a simple example is now considered for the case of a square domain with four corner supports, see Figure 6.

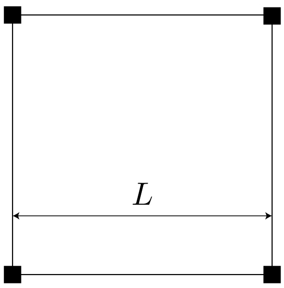


Figure 6: Square domain with corner supports, subject to uniformly distributed loading.

3.1.1. 'Member adding' procedure

The adaptive 'member adding' procedure is shown in Figure 7(a): in the first iteration, problem (10) is solved using a small subset of members (1,056 of 25,456 members in the full 'ground structure'). This first iteration shows a cross-shaped crease in the vault spanning to the corner supports with high forces, and then with secondary parallel spanning arches. Then inequality condition (B.2) is checked to identify violated members that are not present in the subset; and the 244 most violated members are activated, then problem (10) is solved with the new subset, improving the objective value (i.e., volume) from $0.9034pL^3/\sigma$ to $0.8974pL^3/\sigma$ to give the second iteration result Figure 7(b). This process repeats until constraint (B.2) is satisfied for all inactivated members in the 'ground structure', so that no member can be added to further improve the solution. The final iteration (Figure 7(c)) contains only 1,620 active members, while the solution is mathematically guaranteed (as described in Appendix B) to be equivalent to that of solving the full problem with 25,456 members. Therefore, the 'member adding' approach significantly improves computational efficiency, permitting very large scale problems to be solved. The final result shows a vault with bands of diagonal forces travelling directly to the supports, accompanied again by the parallel arches spanning the free edges direction. Note that intersecting members in the 2D 'ground structure' may not intersect in 3D. This limitation may affect the method's practicality and warrants further research, although it is notable that this issue does not arise in even the lower resolution results shown in this paper. When a dense nodal grid is utilized in the 'ground structure,' the resulting vault will be single-layered, as outlined in Proposition 6.4 of [32].

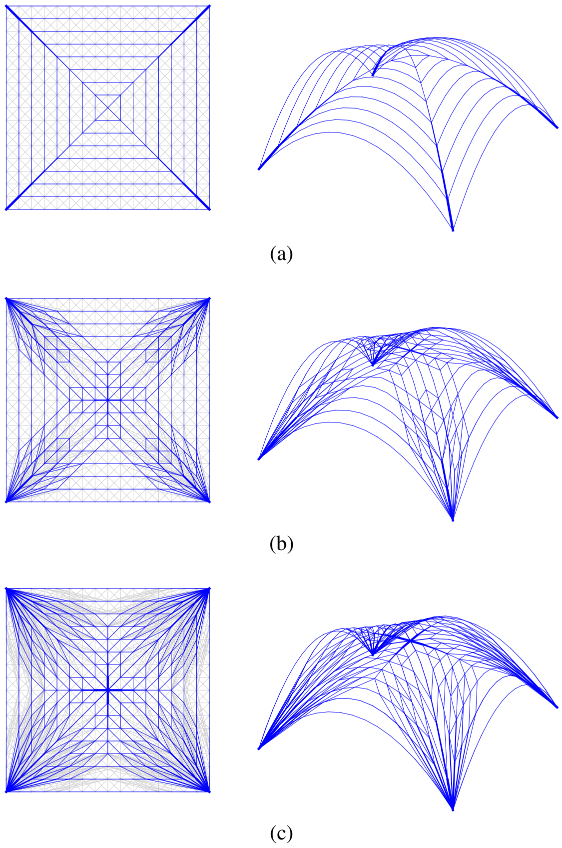


Figure 7: Vault supported at four corners: full problem has 25,456 potential members; adaptive solution process: a) iteration 1, $V_1 = 0.9034\ pL^3/\sigma$, m = 1,056; b) iteration 2, $V_2 = 0.8974\ pL^3/\sigma$, m = 1,300; c) final iteration, $V = 0.8900\ pL^3/\sigma$, m = 1,620 (m is the number of 'activated' members).

3.1.2. Theoretical minimum material designs

Since problem (10) is convex, global optimum solutions can be guaranteed. Therefore, the theoretically minimum material designs can be 'obtained' when the number of nodes in the 'ground structure' tends to infinity. A numerical means of predicting the theoretical solution is via extrapolation; following work in [30, 27], numerical solutions obtained from layout optimization appear to follow a relation of the form

$$V_n = V_\infty + kn^{-\alpha}, \tag{19}$$

where V_n is the volume for problem with n nodal divisions, V_∞ is the volume for $n \to \infty$, and k and α are constants.

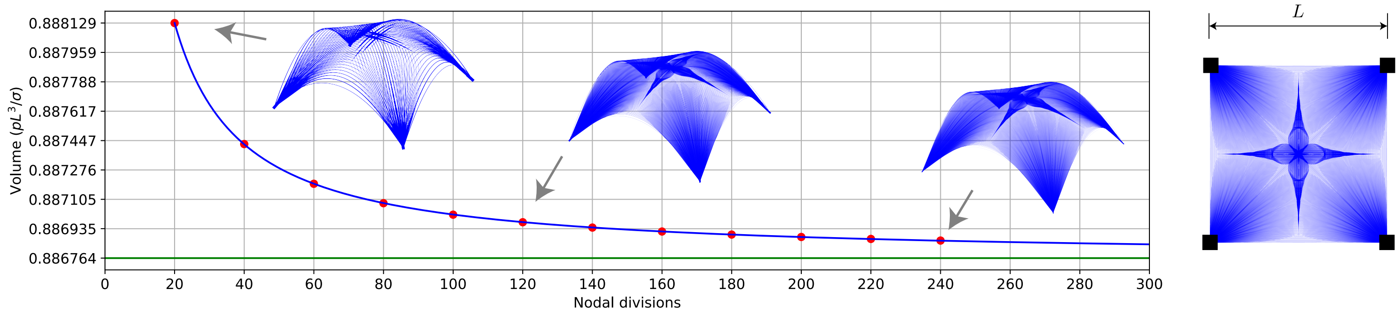


Figure 8: Square domain with corner supports: Optimal volumes obtained at different nodal densities, with extrapolation to estimate the theoretical minimum material of an infinite resolution problem via convergence study.

Table 1: Square domain with corner supports: effects of varying nodal density on the normalized optimal volume. Percentage differences of the results between the extrapolated optimal volume shown, alongside CPU analysis times.

ndiv	n	m	active members	V^* (pL^3/σ)	diff† %	iterations	CPU‡ (s)
10	121	4,492	873	0.88946	0.30	4	0.9
20	441	59,456	4,593	0.88813	0.15	5	2.4
40	1,681	859,168	31,183	0.88743	0.075	6	22
60	3,721	4,209,056	118,828	0.88720	0.049	8	155
80	6,561	13,088,448	331,598	0.88708	0.036	8	474
100	10,201	31,627,760	754,337	0.88702	0.028	9	1,427
120	14,641	65,169,960	1,528,421	0.88697	0.024	10	3,565
140	19,881	120,159,272	2,777,077	0.88694	0.020	10	7,111
160	25,921	204,238,120	4,691,970	0.88692	0.017	11	14,289
180	32,761	326,238,808	7,373,910	0.88690	0.015	11	24,555
200	40,401	496,179,952	11,084,004	0.88689	0.014	12	45,250
220	48,841	725,132,192	16,084,649	0.88688	0.013	11	64,609
240	58,081	1,025,371,872	22,904,142	0.88687	0.012	14	137,413
∞	∞	∞	-	**0.8868**	0	-	-

$*$: solved using quarter design domain due to symmetry, though volume of the whole structure is quoted
$\dagger$: percentage difference compared to the extrapolated solution
$\ddagger$: total CPU cost in the full 'member adding' process , i.e., solving problem (10) via Mosek and checking (B.2)

The extrapolation scheme is applied to the square vault example in Figure 6, and the results shown in Figure 8. In this plot, the normalized optimal volume is plotted against increasing nodal divisions (up until 240), with numerical data results drawn in red markers, and the extrapolation function plotted in blue, tending to the (infinite) limit nodal division result plotted by the horizontal green line. It can be observed that the solutions obtained using high resolution nodal grids resemble the so-called Michell continua comprising infinitesimally small members. The mathematical theory of these continual vaults is a part of the work [32].

Table 1 summarises all solutions for the problem in Figure 6. It is demonstrated that layout optimization can obtain highly accurate solutions even with relatively coarse nodal grids. For example, the solution with 10×10 nodal divisions is only 0.30% above the extrapolated theoretical optimum, owing to the vast number of potential connectivities provided by the 'ground structure. It is therefore possible to utilise the method to generate more practical designs while maintaining structural efficiency, as demonstrated in the examples presented in the following sections. On the other hand, the method is capable of solving enormously large problems containing $> 1,000,000,000$ potential members. The efficacy and robustness of the method are hereby demonstrated.

3.2. Square domain with edge supports

Since layout optimization is capable of generating (near-)optimum solutions, they can be used as benchmarks to evaluate the optimality status of designs generated via classical form-finding methods. For sake of simplicity, the FDM is considered here as a comparative method.

The first comparison example considers a square domain with edge supports. Using layout optimization (LO), a benchmark design is obtained using the aforementioned extrapolation approach and the optimum volume is found to be $0.435806pL^3/\sigma$, see Figure 9. Then the example is solved using the FDM with a predefined orthogonal grid connectivity, see Figure 10. To identify the optimum constant force density across all members, the following linear search problem is solved,

$$\min_{\gamma} V(\gamma \mathbf{f}_\mathrm{d}), \tag{20}$$

where $\mathbf{f}_\mathrm{d}$ is a vector containing the initial force density set to all members. Scalar γ is a multiplier to be optimized for minimum volume. In this paper, the initial force density is set to $1.0pL$, and γ varies within the range from 0.0001 to 10000.

The minimum material design via FDM is shown in Figure 10, which is 18.2% above the benchmark, i.e., the extrapolated theoretical minimum material design in Figure 9. In addition, three further different predefined connectivities are considered, see Figure 11. These additional connectivities for study are: #2 orthogonal plus central emanating diagonals, #3

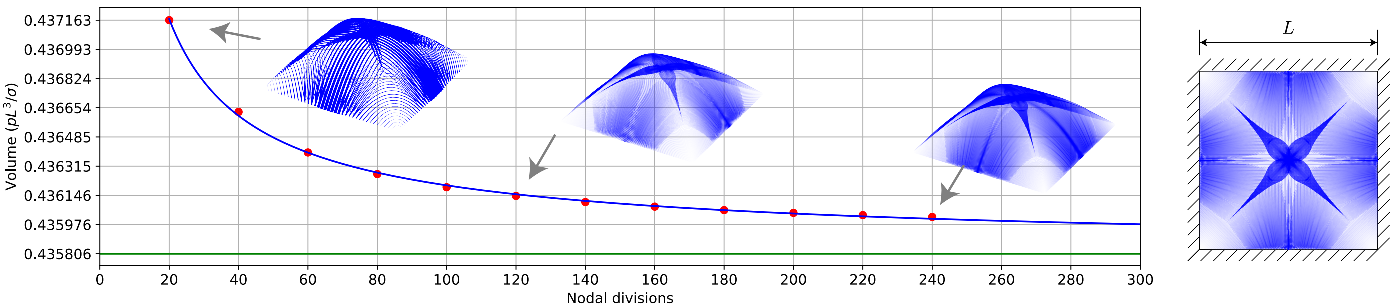


Figure 9: Square domain with edge supports: Optimal volumes obtained at different nodal densities, with extrapolation to estimate the theoretical minimum material of an infinite resolution problem via convergence study.

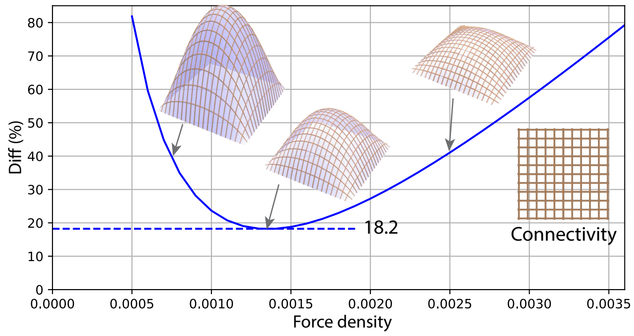


Figure 10: Force Density Method: identifying the optimum uniform force density for the edge supported square vault, by solving a single variable linear-search problem (20)

orthogonal plus concentric diamond diagonals, and #4 orthogonal plus superimposed cross diagonals. It can be observed that, for the FDM results, although the resulting vault geometric forms for the different connectivities are similar to each other, the volumes exhibit significant variations, even when the uniform force densities are optimized. Specifically, connectivity #2 leads to a large increase in volume (31%), while connectivity #3 shows the least (13.8%) increase over the benchmark case. Note that different connectivities or topological mappings [22] may be proposed, each with varying differences in volume compared to the benchmark.

To compare the solutions to those obtained via layout optimization, two cases are considered: first, similar to the FDM, the four predefined connectivities are used directly; second, the standard layout optimization problem (i.e., no predefined connectivity) is solved. In the first case (see again Figure 11), it is found that connectivity #1 leads to the largest volume increase (4.99%), since it has the least design freedom (i.e., the least amount of design variables). It is worth noting that with this basic connectivity the generated structure coincides with the archgrid solution [34], see Proposition 8.6 in [32] for the proof. Similar to the FDM, connectivity #3 gives the best design (1.30% increase). Nevertheless, these results are significantly lower than those generated via the FDM with constant force densities, as each member can have an optimized force value in the LO method, rather than constant force (density) values across all members. In the second case, the solution obtained with full connectivity is shown in Figure 11. Due to the increased design freedom, this leads to the most efficient design, albeit with increased complexity.

3.3. Six-sided domain supported at corners

The second comparison example considers a six-sided domain with corner supports, as shown in Figure 12(a) with two layouts of the predefined connectivity, Figure 12(b) is used in the the FDM and Figure 12(c) for LO. To consider alternative designs with different force density distributions beyond just using constant values throughout, two special cases are introduced by reinforcing the edges and ribs, by adjusting the relative force densities of the corresponding members to $10.0pL$ (thicker lines), while maintaining the remaining members at a force density of $1.0pL$ (thinner lines), see Figure 13. The results of the FDM analyses are shown in Figure 14(a-c), where it can be observed that the FDM pulls the edge members inwards by adjusting positions of free nodes in plane; and that such effect is reduced when the edges are reinforced with the higher force densities, effectively pulling the structure back out. Since the loaded area of the UDL (i.e., the projected vault area on the xy plane) is altered, the magnitude of equivalent point loads should be updated accordingly. This therefore results in an iterative process, where the FDM is solved sequentially with updated equivalent point loads, until the change in volume between two successive iterations is sufficiently small.

In the layout optimization approach, three cases are considered by varying the potential members in the 'ground structure'. The least material design is obtained first in Figure 14(d), where a fully connected 'ground structure' is employed. The reported volume is then used as the benchmark to assess the efficiency of all other designs. Next, Figure 14(e) shows the solution obtained with the adjacent connectivity in Figure 12(c), where the solution is only 0.2% above the benchmark case. While, in Figure 14(f), the same connectivity in Figure 12(b) is used, leading to a slight increase in volume (2.5%). Also note that, in layout optimization, the boundaries of the design domain remain preserved, eliminating the necessity for an iterative process to update the equivalent point loads employed in FDM.

Since the plane area of vaults is altered in the FDM, the volume per area, $\bar{V}$, is used to compare the results in Figure 14. Regarding the FDM solutions, using a uniform force density in Figure 14(a), leads to the most inefficient design, where a volume increase of 34.9% is reported. Strengthening the edges

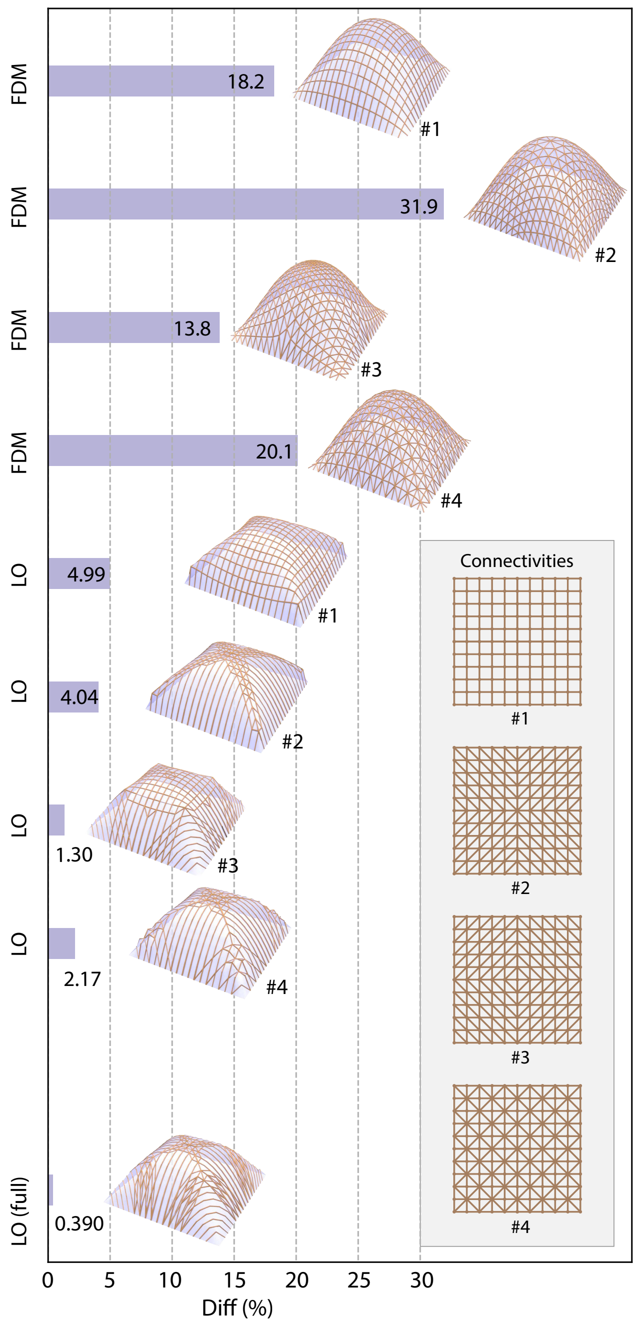


Figure 11: Comparative study by varying initial connectivities in FDM and layout optimization (OPT). Note that for clarity, a 10×10 nodal grid is used to illustrate the connectivities, though the problems are solved using a 20×20 nodal grid.

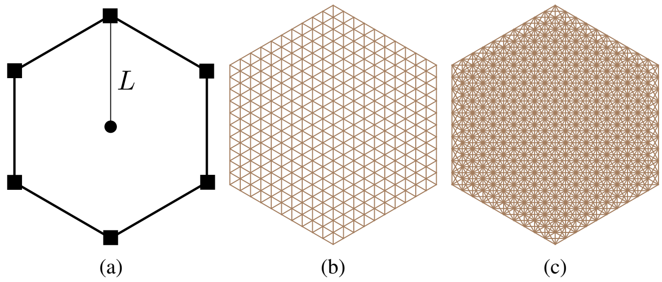


Figure 12: Six-sided problem: (a) problem specification; (b) minimum connectivity for FDM; (c) adjacent connectivity for layout optimization.

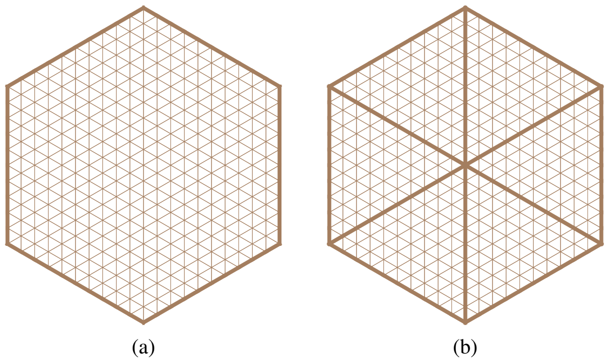


Figure 13: Six-sided problem, special cases in FDM: (a) edges have force densities of $f_{\mathrm{d}} = 10.0pL$; (b) similar to (a), but with ribs additionally reinforced.

and ribs will improve its structural efficiency, see Figure 14(b) and (c). Further improvements can be achieved by introducing additional variations or optimizations of the force densities, albeit it necessitates further exploration and approaches. On the contrary, solutions obtained via layout optimization exhibit only minor increases in volume over the benchmark value, even though the member layouts may vary significantly (e.g., compare Figure 14(e) to (f)).

3.4. Eight-sided problem under UDL and a central point load

The last comparison example considers an eight-sided domain with edge supports around the perimeter. The vault is subject to a UDL and a centrally loaded point load with a magnitude of $F = \lambda pL^2$, where p is the magnitude of the UDL, λ is a predefined multiplier, and L is the distance from centre to perimeter vertices (i.e., radius of the circumcircle), see Figure 15(a).

For the FDM, the connectivity in Figure 15(b) is used, with a special case in Figure 15(c), where the ribs are reinforced by setting the force densities to $10.0pL$, similar to the previous example in Figure 13. For layout optimization, two cases are considered: one with a fully connected ‘ground structure’ and the other using the connectivity shown in Figure 15(b). To study the influence of the point load, three scenarios are considered,

- $\lambda = 0$: only the UDL is present,
- $\lambda = 0.1$: both the UDL and point load are present together,
- $\lambda = 1$: a relatively large point load is used.

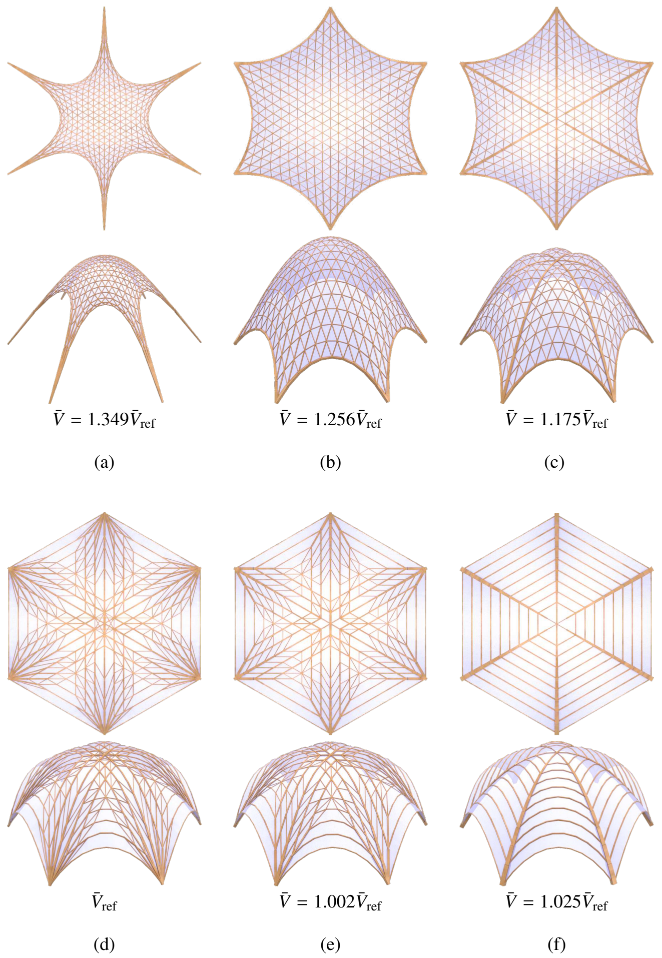


Figure 14: Six-sided domain supported at corners: (a) FDM, $\bar{V} = 1.349\bar{V}_{\mathrm{ref}}$; (b) FDM with reinforced edges, $\bar{V} = 1.256\bar{V}_{\mathrm{ref}}$; (c) FDM with reinforced edges and ribs, $\bar{V} = 1.175\bar{V}_{\mathrm{ref}}$; (d) LO with full connectivity, $\bar{V}_{\mathrm{ref}}$; (e) LO with adjacent connectivity, $\bar{V} = 1.002\bar{V}_{\mathrm{ref}}$; (f) LO with minimum connectivity, $\bar{V} = 1.025\bar{V}_{\mathrm{ref}}$ ($\bar{V}$ represents the material volume per area on plane)

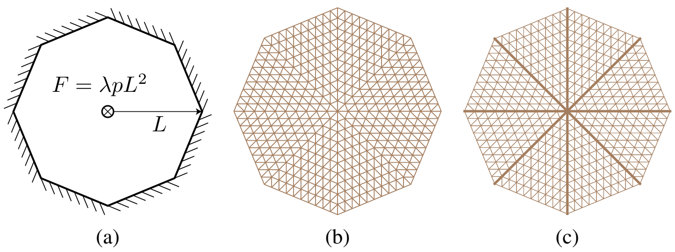


Figure 15: Eight-sided problem: (a) problem specification; (b) connectivity in FDM; (c) with reinforced ribs.

The results are shown in Figure 16, for the two FDM cases (a) and (b), the two LO cases (c) and (d), and for the three λ loadcases. The solutions in (c), which are obtained with a fully connected 'ground structure', are used as the benchmarks volumes under each scenario ($\lambda = 0, 0.1, 1$) to compare to. The standard FDM with uniform force densities in (a) leads to the largest volume increase compared with the benchmark volumes, followed then by designs (b) and (d). As the magnitude of the point load increases, the solutions derived from the FDM and layout optimization methods evolve differently. The standard FDM in Figure 16(a) develops a localized spike at the tip of the vault under the point load, while the optimized solutions in (c) and (d) evolve towards more pyramid-shaped vaults with more defined creases. Reinforcing the ribs with the FDM effectively transforms the shape, also showing an evolution towards pyramid shapes; and effectively reducing the structural volume.

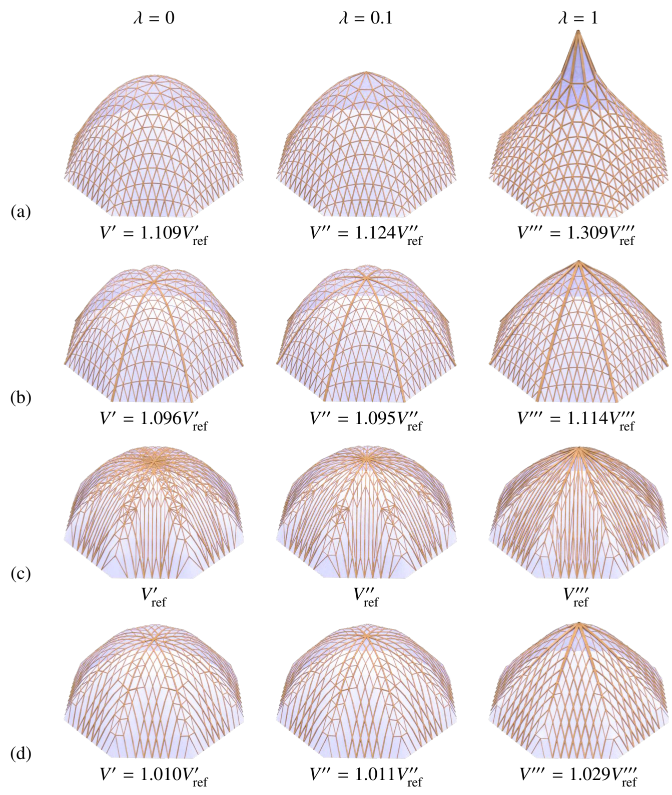


Figure 16: Eight-sided vault with centroid point load: (a) FDM; (b) FDM with reinforced ribs; (c) LO with a full 'ground structure'; (d) LO with minimum connectivity

3.5. *Summary of comparison examples*

To further demonstrate the efficacy of the layout optimization form-finding approach, the CFDM from [21] , which allows for variable force densities to obtain optimum (minimum volume) designs, was also applied for comparison. Table 2 summarises all results and CPU costs associated with the different methods, FDM, CFDM and LO. It can be observed that although the CFDM obtains the same results, the associated CPU costs are relatively higher, due to the use of slower non-linear optimization solvers. This also makes it challenging to solve larger problems such as that in Figure 12(c), which led to an error during to optimization process. Similar to LO, CFDM can effectively restrict the horizontal movement of boundary nodes in the example shown in Figure 12, thereby preserving the plane area of the vaults. This contrasts with the result obtained via FDM, where the distribution of UDL is altered, resulting in higher volume per unit area, as observed in Figure 14.

It can also be observed from Table 2 that neither FDM nor CFDM can deal with the full problem, thus a predefined connectivity is always required. This limitation is now addressed in LO.

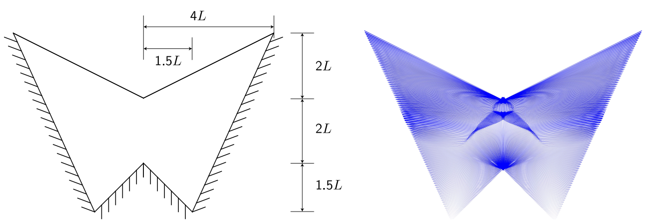
Table 2: Comparing outcome volumes and CPU costs associated with the FDM, CFDM and LO method.

	FDM		CFDM		LO	
Problem	V (pL^3/σ)	CPU† (s)	V	CPU‡	V	CPU*
Fig. 11*, #1	0.51472	0.5	0.45732	280.3	0.45732	0.1
Fig. 11*, #2	0.57468	0.5	0.45320	566.6	0.45320	0.1
Fig. 11*, #3	0.49568	0.5	0.44127	582.6	0.44127	0.1
Fig. 11*, #4	0.52336	0.5	0.44506	559.0	0.44506	0.1
Fig. 11*, full	-	-	-	-	0.43730	0.3
Fig. 12(b)	1.1603	2.5	2.9943	621.0	2.9943	0.3
Fig. 12(c)	-	-	-	-	2.9270	0.3
Fig. 12, full	-	-	-	-	2.9206	1.3
Fig. 15, $\lambda = 0$	2.6081	2.5	2.3741	997.1	2.3741	0.3
Fig. 15, $\lambda = 1$	5.2027	2.5	4.0876	1266	4.0876	0.3

$*$: solved using quarter design domain due to symmetry, though volume of the whole structure is quoted
$\dagger$: total CPU cost in linear search problem (20)
$\ddagger$: total CPU cost for non-linear solver IPOPT
$\star$: total CPU cost in the full 'member adding' process , i.e., solving problem (10) via Mosek and checking (B.2)

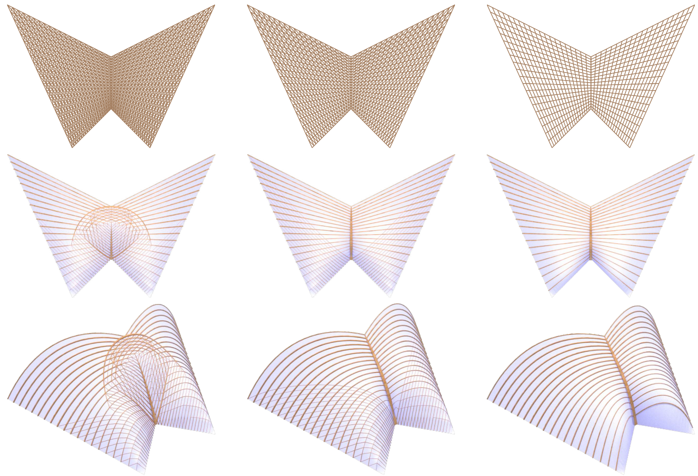
3.6. Irregular domain

To further demonstrate the layout optimization method, a problem with an irregular domain is considered here. The 'ground structure' approach eliminates the need of an initial guess of the mesh pattern or member connectivities. The benchmark solution is obtained by employing a fine 'ground structure' containing $> 10,000$ nodes and $> 30,000,000$ potential members, see Figure 17. Similar to the solutions in Figure 8, the outcome resemble the Michell continua comprising infinitesimally small members. To obtain more rational results, different connectivies shown in Figure 18 are used; and the differences to the benchmark solution remain relatively low.



(a) (b)

Figure 17: Irregular domain: benchmark solution obtained with >10,000 nodes and >30,000,000 potential members: (a) problem dimension; (b) optimum design, $V = V_0$



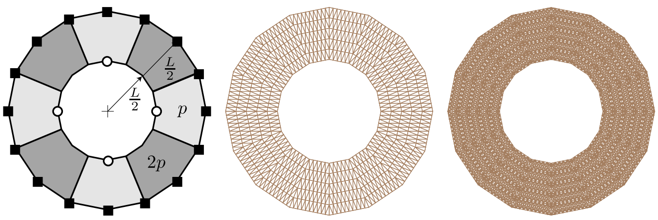
(a) (b) (c)

Figure 18: Irregular domain, various designs obtained using different connectivities: (a) fine triangulation, $V = 1.044V_0$; (b) relatively coarse triangulation, $V = 1.069V_0$; (c) structured quad mesh, $V = 1.093V_0$

3.7. Domain with hole under patch loads

The final example involves a domain with a hole subjected to patch loads of magnitudes p and $2p$, as illustrated in Figure 19(a). Three design scenarios for the hole are considered:

- free, where the hole is unsupported;
- roller-supported, where the hole is supported by four rollers positioned as shown in Figure 19(a);
- fixed-supported, where the hole is fixed supported at four anchor points.



(a) (b) (c)

Figure 19: Domain with hole under patch loads and internal supports: (a) problem specification, where all light and dark gray areas are under UDL with magnitudes of p and $2p$, respectively; circular symbol represents internal support with three design scenarios: free, roller-supported, or fixed-supported; (b) minimum connectivity; (c) adjacent connectivity.

The results for the three design scenarios and different types of connectivity are shown in Figure 20. When comparing the different types of supports, it is evident that the structural volume decreases as stronger support conditions are applied. When comparing the different types of connectivity, it is interesting to observe that the extent of volume reduction achieved by using adjacent connectivity instead of minimum connectivity varies depending on the support type. In Figure 20(a), the volume reduction is relatively small, indicating that the design obtained using minimum connectivity remains an efficient solution. However, in Figure 20(b) and (c), the reduction becomes

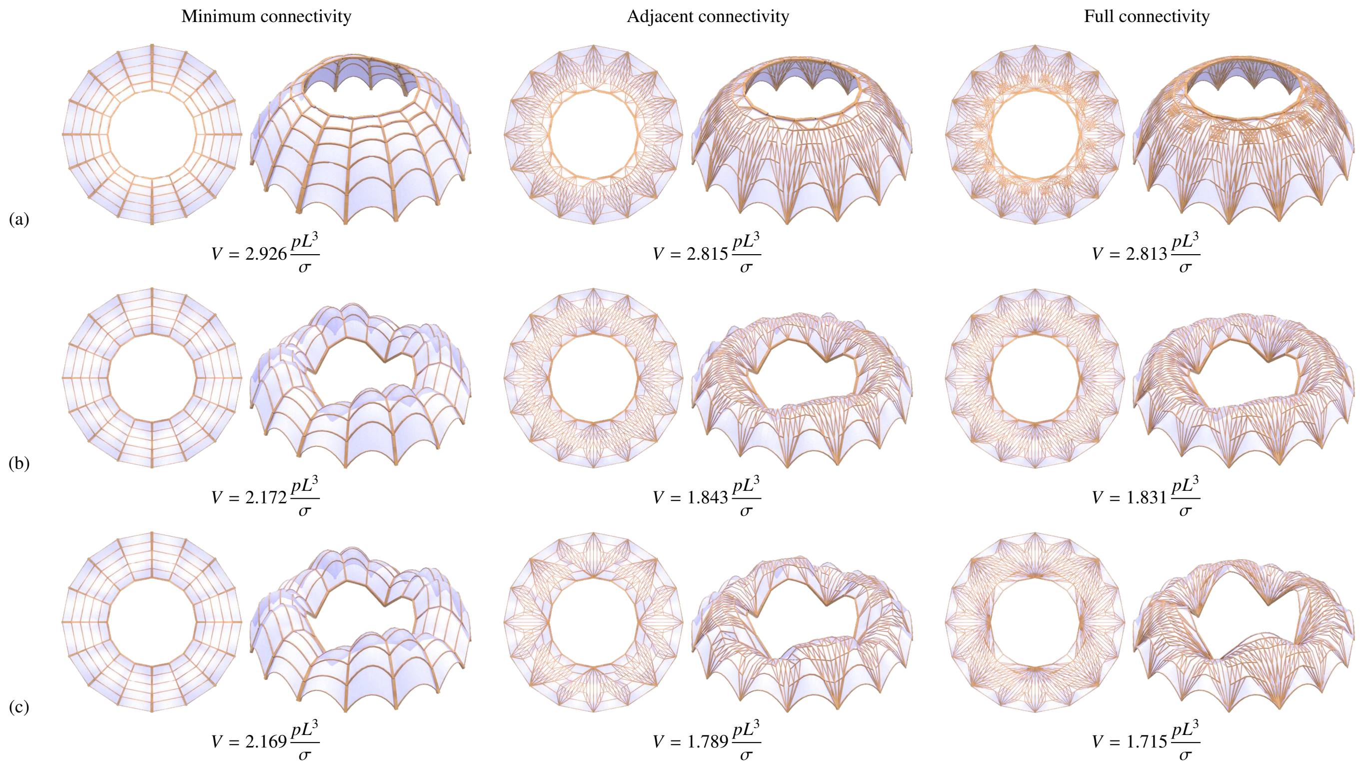


Figure 20: Domain with hole under patch loads, considering three scenarios for the internal supports in Figure 19: a) free; b) roller-supported; c) fixed-supported

significantly larger, as the additional members introduced by the adjacent connectivity enable the formation of more efficient structural configurations for transferring the load to the point supports. This highlights the advantages of employing the method in more complex design scenarios, particularly when defining a predefined connectivity based on the designer's experience is challenging. In addition to identifying the optimum structural volume, the method's ability to automatically generate the optimum force and form can also inspire designers to develop more practical designs or connectivity patterns for use in other form-finding approaches.

4. Conclusions

Although many form-finding methods have been developed in recent decades to identify bending-free designs of vaults, their reliance on predefined layouts of the underlying force flow, or of simplified distributions of these forces, can result in designs that are materially inefficient. These methods can also have disadvantages such as drifting nodes in plan and long optimization times for non-trivial internal force distributions. In this contribution, a numerical layout optimization procedure tailored for vault form-finding problems has been described, building on the work by Bołbotowski [32]. Although the underlying optimization problem remains the same as in [32], an engineering perspective is employed here to offer an alternative approach that is significantly more accessible to the community. The method uses the 'ground structure' approach, formulating the form-finding problem as a two-dimensional problem solvable via conic programming. Since the optimization problem is convex, globally optimal solutions can be obtained, which means that the resulting vaults are guaranteed minimum material designs. In addition, an adaptive 'member adding' technique has been used to significantly improve computational efficiency, enabling medium-sized problems to be solved in seconds and also allowing very high-precision solutions to be obtained for benchmarking purposes (e.g., problems comprising over 1,000,000,000 design variables can be solved on a desktop PC). The proposed procedure has been applied to a range of example problems previously solved using traditional form-finding methods, namely using the FDM, demonstrating the ability of the proposed procedure to generate designs that are significantly more materially efficient in every case. It is shown that simple modifications, such as restricting connectivies in the 'ground structure', can generate more practical designs with only slight increase of material volume. The presented method can be used to design the guaranteed most materially efficient vaults with general plan domain and supports, and with fixed projection of the nodes consistent with the applied loading.

Data availability

Downloadable Python script and example files are available from: `https://doi.org/10.15131/shef.data.27187602`. Note that only free libraries are included in the script, therefore, although the Mosek solver is used in the paper, it is not included in the script. Interested readers can use `vol = prob.solve(solver = cvx.MOSEK)` to activate the solver, provided that Mosek is installed and licensed.

Appendix A. Existence of compatible nodal elevations

With $\mathbf{q}_{xy}$, $\mathbf{q}_z$ being solutions of (10), the question of existence of compatible nodal elevations $\mathbf{z}$ posed in Section 2.5 boils down to the solvability of the system:

$$\mathbf{B}_z^{\mathrm{T}}\mathbf{z} = \mathbf{L}_{xy}\mathbf{Q}_{xy}^{-1}\mathbf{q}_z, \tag{A.1}$$

where $\mathbf{L}_{xy} = \mathrm{diag}(\mathbf{l}_{xy})$, $\mathbf{Q}_{xy} = \mathrm{diag}(\mathbf{q}_{xy})$. All forces q_{xy} are assumed to be positive, otherwise the rows corresponding to zero forces can be removed from the system (A.1).

Recall the form (6) of the objective function in the problem (10), where no variable $\mathbf{r}$ is present. Since the double minimization in $\mathbf{q}_{xy}$, $\mathbf{q}_z$ can be iterated, $\mathbf{q}_z$ must solve the problem,

$$\min_{q_z} \quad \mathbf{q}_z^{\mathrm{T}}\mathbf{L}_{xy}\mathbf{Q}_{xy}^{-1}\mathbf{q}_z \tag{A.2a}$$

$$\text{s.t.} \quad \mathbf{B}_z\mathbf{q}_z = \mathbf{f}_z. \tag{A.2b}$$

It is a problem of minimizing the square of a norm on an affine subspace determined by the linear system (A.2b). The norm comes from the scalar product $\mathbf{x} \cdot \mathbf{y} = \mathbf{x}^{\mathrm{T}}(\mathbf{L}_{xy}\mathbf{Q}_{xy}^{-1})\mathbf{y}$. By the orthogonal projection theorem the solution $\mathbf{q}_z$ is orthogonal to the null space of the matrix $\mathbf{B}_z$ with respect to this very scalar product. By a standard result in linear algebra this means that $\mathbf{q}_z$ lies in the image of the adjoint of $\mathbf{B}_z$. With the chosen scalar product this adjoint equals $\mathbf{Q}_{xy}\mathbf{L}_{xy}^{-1}\mathbf{B}_z^{\mathrm{T}}$. Therefore, the equation $\mathbf{Q}_{xy}\mathbf{L}_{xy}^{-1}\mathbf{B}_z^{\mathrm{T}}\mathbf{z} = \mathbf{q}_z$ has a solution, and so does (A.1).

Appendix B. Dual formulation and 'member adding'

Appendix B.1. Dual Formulation

According to duality theory ([39]), the dual problem of (10) can be written as:

$$\max_{u_{xy},u_z,t_1,t_2,t_3} \quad W = \mathbf{f}_z^{\mathrm{T}}\mathbf{u}_z \tag{B.1a}$$

$$\text{s.t.} \quad \mathbf{t}_2 + \mathbf{B}_{xy}^{\mathrm{T}}\mathbf{u}_{xy} = \frac{1}{\sigma}\mathbf{l}_{xy} \tag{B.1b}$$

$$\mathbf{t}_3 + \mathbf{B}_z^{\mathrm{T}}\mathbf{u}_z = \mathbf{0} \tag{B.1c}$$

$$\mathbf{t}_1 = 2\frac{1}{\sigma}\mathbf{l}_{xy} \tag{B.1d}$$

$$2t_1t_2 \geq t_3^2, \quad \text{for all members} \tag{B.1e}$$

where $\mathbf{u}_z$ and $\mathbf{u}_{xy}$ are the virtual displacement variables of each node in vertical and horizontal directions, respectively. $\mathbf{t}_1$, $\mathbf{t}_2$ and $\mathbf{t}_3$ are auxiliary variables used to express the problem in standard form. Also, without going into details, it can be proven that constraint (9) satisfied the so-called constraint qualifications, more specifically, Slater's condition. Interested readers may refer to [39] for details. This ensures 'strong duality', which means that the solution of (B.1) is also that of (10).

Appendix B.2. Adaptive 'member adding'

Constraints in (B.1) can be reformulated by eliminating variables $\mathbf{t}_1$, $\mathbf{t}_2$ and $\mathbf{t}_3$ in the equalities:

$$2\left(2\frac{1}{\sigma}l_{xy}\right)\left(\frac{1}{\sigma}l_{xy} - \left[\mathbf{B}_{xy}^{\mathrm{T}}\mathbf{u}_{xy}\right]_i\right) \geq \left(\left[\mathbf{B}_z^{\mathrm{T}}\mathbf{u}_z\right]_i\right)^2, \tag{B.2}$$

where the notation $[\ \ ]_i$ indicates that once the vector inside the brackets has been calculated, the i'th element (i.e. corresponding to the current element i) should be extracted.

Note that modern convex optimization solvers provide the solutions to both (10) and (B.1) whenever either is solved, and therefore the displacement variables $\mathbf{u}_{xy}$ and $\mathbf{u}_z$ can be easily obtained when (10) is solved. Since $\mathbf{u}_{xy}$ and $\mathbf{u}_z$ are virtual displacements of nodes, they are available without requiring all members in (10). The critical observation which allows the adaptive 'member adding' method to be understood, is that constraint (B.2) can be calculated for any member, regardless of whether it was activated or not. While constraint (B.2) is always satisfied for activated members, violations may occur for those inactivated. This therefore provides a means of identifying new members to be activated in the next iteration in order to satisfy (B.2). It is observed that a large number of inactivated members may violate constraint (B.2) initially; to improve computational efficiency, only the most violated members are activated in each iteration. Interested readers may refer to [38] for details. If constraint (B.2) is satisfied for all members, (B.1) is solved for the full problem; therefore, according to the aforementioned 'strong duality', the solution of (10) for the full problem is found.

Appendix B.3. Infeasible sub-problems

In the 'member-adding' procedure, the initial connectivity may not always satisfy force equilibrium conditions, potentially resulting in infeasible optimization problems. To address this, additional members from the full 'ground structure' can be activated until feasibility is restored. As explained in Appendix B.1 and Appendix B.2, since the 'member-adding' procedure is guaranteed to obtain the same solution as the full problem, various heuristic methods can be developed without compromising the rigour of the layout optimization method.

Here, a simple approach similar to that discussed in [40] is adopted. When the primal problem (10) is infeasible, the dual problem (B.1) becomes unbounded, indicating that at least one node exhibits infinite virtual displacement (i.e., $|u_z| \to \infty$). In addition to reporting the infeasibility status, solvers such as Mosek will still provide dual variables for diagnostic purposes. This allows constraint (B.2) to be utilized for identifying members to be added to restore feasibility. Potential members connected to a node with near-infinite virtual displacement will lead to significant constraint violations and are therefore prioritized for addition.

Figure B.21 shows the solutions for the vault problem in Figure 6 using different initial connectivities. Although varying starting connectivity results in different intermediate results, the final solutions remain identical.

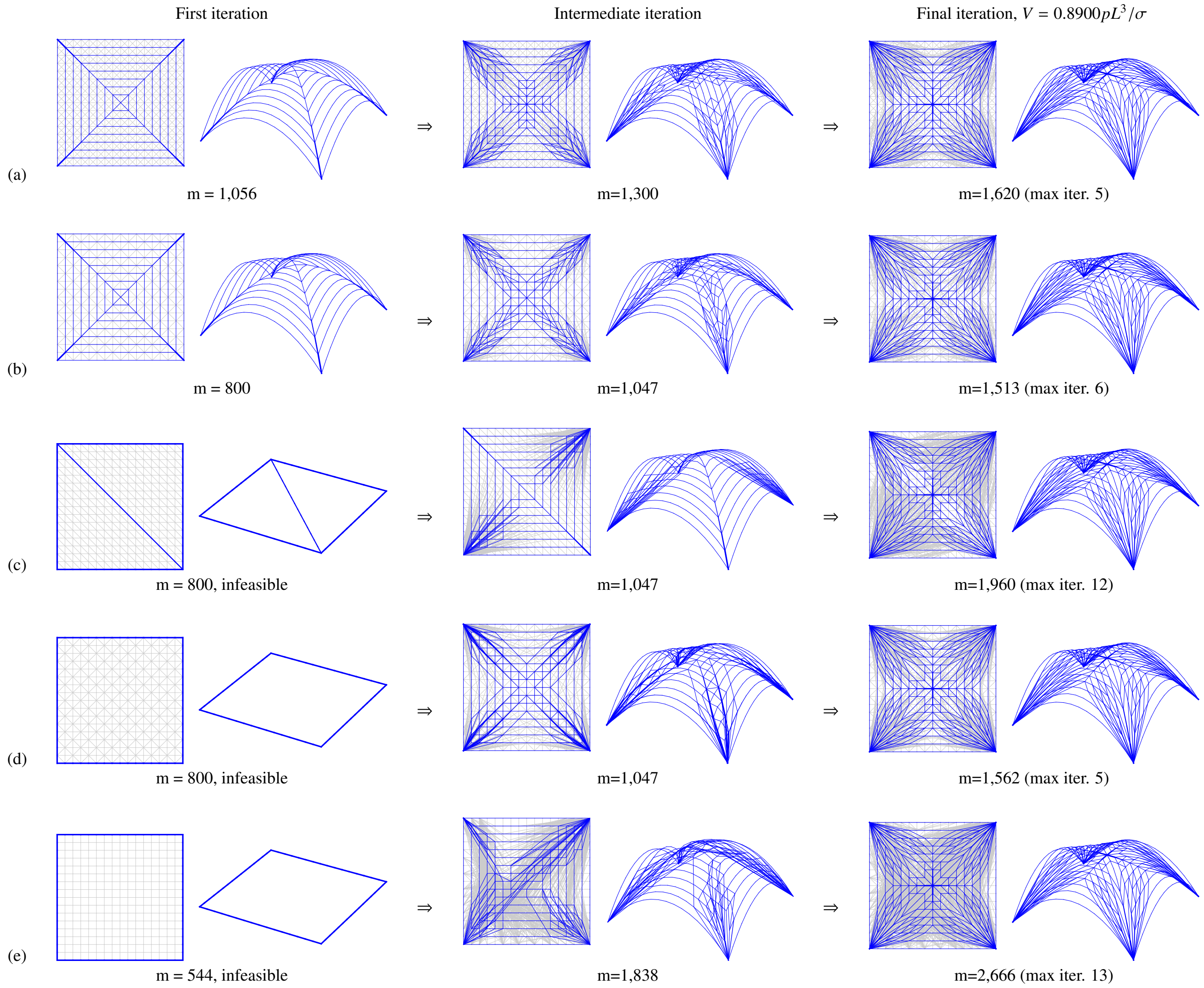


Figure B.21: Vault supported at four corners: varying starting connectivities in the 'member adding' procedure: a) adjacent connectivity; b) adjacent connectivity without crossing members; c) - e) alternative connectivites leading to infeasible starting sub-problems. Note that although the member subsets in the ground structure of the final iteration are different, the resulting layouts and the reported structural volume remain identical (full 'ground structure' contains 25,456 members)

References

[1] S. Miccoli, L. M. Gil-Martín, and E. Hernández-Montes, "New historical records about the construction of the Arch of Ctesiphon and their impact on the history of structural engineering," *Notes and Records: the Royal Society Journal of the History of Science*, vol. 77, pp. 113–134, Feb. 2023.

[2] S. Adriaenssens, P. Block, D. Veenendaal, and C. Williams, eds., *Shell structures for architecture: form finding and optimization*. London ; New York: Routledge/ Taylor & Francis Group, 2014.

[3] D. Veenendaal and P. Block, "An overview and comparison of structural form finding methods for general networks," *International Journal of Solids and Structures*, vol. 49, no. 26, pp. 3741–3753, 2012.

[4] A. Siev and J. Eidelman, "Stress analysis of prestressed suspended roofs," *Journal of the Structural Division*, vol. 90, no. 4, pp. 103–121, 1964.

[5] H.-J. Schek, "The force density method for form finding and computation of general networks," *Computer Methods in Applied Mechanics and Engineering*, vol. 3, no. 1, pp. 115–134, 1974.

[6] K. Linkwitz and H. J. Schek, "Einige bemerkungen zur berechnung von vorgespannten seilnetzkonstruktionen," *Ingenieur-archiv*, vol. 40, pp. 145–158, 1971.

[7] K. Linkwitz, "About formfinding of double-curved structures," *Engineering Structures*, vol. 21, no. 8, pp. 709–718, 1999.

[8] P. Block and J. Ochsendorf, "Thrust network analysis: a new methodology for three-dimensional equilibrium," *Journal of the International Association for Shell and Spatial Structures*, vol. 48, no. 3, pp. 167–173, 2007.

[9] L. Lachauer and P. Block, "Interactive equilibrium modelling," *International Journal of Space Structures*, vol. 29, no. 1, pp. 25–37, 2014.

[10] D. O'Dwyer, "Funicular analysis of masonry vaults," *Computers & Structures*, vol. 73, no. 1-5, pp. 187–197, 1999.

[11] F. Fraternali, "A thrust network approach to the equilibrium problem of unreinforced masonry vaults via polyhedral stress functions," *Mechanics Research Communications*, vol. 37, no. 2, pp. 198–204, 2010.

[12] F. Marmo and L. Rosati, "Reformulation and extension of the thrust network analysis," *Computers & Structures*, vol. 182, pp. 104–118, 2017.

[13] M. Barnes, *Form finding and analysis of tension space structures by dynamic relaxation*. PhD thesis, City University London, UK, 1977.

[14] M. R. Barnes, "Form-finding and analysis of prestressed nets and membranes," *Computers & Structures*, vol. 30, no. 3, pp. 685–695, 1988.

[15] M. R. Barnes, "Form finding and analysis of tension structures by dynamic relaxation," *International journal of space structures*, vol. 14, no. 2, pp. 89–104, 1999.

[16] A. Kilian and J. Ochsendorf, "Particle-spring system for structural form finding," *Journal of the International Association for Shell and Spacial Structures*, vol. 46, no. 147, 2005.

[17] A. Liew, D. Pagonakis, T. Van Mele, and P. Block, "Load-path optimisation of funicular networks," *Meccanica*, vol. 53, no. 1, pp. 279–294, 2018.

[18] R. Avelino, "Optimising the load path of compression-only thrust networks through independent sets," *Structural and Multidisciplinary Optimization*, vol. 60, pp. 231–244, 2019.

[19] R. M. Avelino, "Assessing the safety of vaulted masonry structures using thrust network analysis," *Computers & Structures*, vol. 257, p. 106647, 2021.

[20] N. A. Nodargi and P. Bisegna, "Generalized thrust network analysis for the safety assessment of vaulted masonry structures," *Engineering Structures*, vol. 270, p. 114878, Nov. 2022.

[21] A. Liew, "Constrained force density method optimisation for compression-only shell structures," in *Structures*, vol. 28, pp. 1845–1856, Elsevier, 2020.

[22] E. Hernández-Montes, R. Jurado-Piña, and E. Bayo, "Topological Mapping for Tension Structures," *Journal of Structural Engineering*, vol. 132, pp. 970–977, June 2006.

[23] A. Moskaleva, S. Gusev, S. Konev, I. Sergeichev, A. Safonov, and E. Hernandez-Montes, "Composite freeform shell structures: Design, construction and testing," *Composite Structures*, vol. 306, p. 116603, Feb. 2023.

[24] W. Dorn, R. Gomery, and H. Greenberg, "Automatic design of optimal structures," *J. de Mecanique*, vol. 3, 1964.

[25] H. Lu, A. Tyas, M. Gilbert, and A. V. Pichugin, "On transmissible load formulations in topology optimization," *Structural and Multidisciplinary Optimization*, vol. 64, pp. 23–37, 2021.

[26] H. E. Fairclough, M. Gilbert, A. V. Pichugin, A. Tyas, and I. Firth, "Theoretically optimal forms for very long-span bridges under gravity loading," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 474, p. 20170726, Sept. 2018.

[27] K. Bołbotowski, L. He, and M. Gilbert, "Design of optimum grillages using layout optimization," *Structural and Multidisciplinary Optimization*, vol. 58, no. 3, pp. 851–868, 2018.

[28] M. Gilbert and A. Tyas, "Layout optimization of large-scale pin-jointed frames," *Engineering Computations*, vol. 20, no. 8, pp. 1044–1064, 2003.

[29] M. Fuchs and E. Moses, "Optimal structural topologies with transmissible loads," *Structural and Multidisciplinary Optimization*, vol. 19, pp. 263–273, 2000.

[30] W. Darwich, M. Gilbert, and A. Tyas, "Optimum structure to carry a uniform load between pinned supports," *Structural and Multidisciplinary Optimization*, vol. 42, no. 1, pp. 33–42, 2010.

[31] Y. Jiang, T. Zegard, W. F. Baker, and G. H. Paulino, "Form-finding of grid-shells using the ground structure and potential energy methods: a comparative study and assessment," *Structural and Multidisciplinary Optimization*, vol. 57, pp. 1187–1211, 2018.

[32] K. Bołbotowski, "Optimal vault problem–form finding through 2d convex program," *Computers & Mathematics with Applications*, vol. 109, pp. 280–324, 2022.

[33] G. I. Rozvany and W. Prager, "A new class of structural optimization problems: optimal archgrids," *Computer methods in applied mechanics and engineering*, vol. 19, no. 1, pp. 127–150, 1979.

[34] R. Czubacki and T. Lewiński, "Optimal archgrids: a variational setting," *Structural and Multidisciplinary Optimization*, vol. 62, no. 3, pp. 1371–1393, 2020.

[35] G. Dzierzanowski and R. Czubacki, "Optimal archgrids spanning rectangular domains," *Computers & Structures*, vol. 242, p. 106371, 2020.

[36] A. G. M. Michell, "The limits of economy of material in frame-structures," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 8, no. 47, pp. 589–597, 1904.

[37] T. Lewiński, T. Sokół, and C. Graczykowski, *Michell Structures*. Springer International Publishing, Cham, 2019.

[38] L. He, M. Gilbert, and X. Song, "A Python script for adaptive layout optimization of trusses," *Structural and Multidisciplinary Optimization*, vol. 60, no. 2, pp. 835–847, 2019.

[39] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge, UK ; New York: Cambridge University Press, 2004.

[40] H. E. Fairclough, L. He, T. B. Asfaha, and S. Rigby, "Adaptive topology optimization of fail-safe truss structures," *Structural and Multidisciplinary Optimization*, vol. 66, p. 148, July 2023.