

Interactive Vault Design

by

M. Rippmann, L. Lachauer, P. Block

Reprinted from

INTERNATIONAL JOURNAL OF SPACE STRUCTURES

Volume 27 · Number 4 · 2012

MULTI-SCIENCE PUBLISHING CO. LTD.

5 Wates Way, Brentwood, Essex CM15 9TB, United Kingdom

Interactive Vault Design

Matthias Rippmann^{1*}, Lorenz Lachauer² and Philippe Block³

¹Research Assistant, Institute of Technology in Architecture, ETH Zurich
Wolfgang-Pauli-Str. 15, 8093 Zurich, Switzerland
rippmann@arch.ethz.ch

²Research Assistant, Institute of Technology in Architecture, ETH Zurich
Wolfgang-Pauli-Str. 15, 8093 Zurich, Switzerland
lachauer@arch.ethz.ch

³Assistant Professor, Institute of Technology in Architecture, ETH Zurich
Wolfgang-Pauli-Str. 15, 8093 Zurich, Switzerland
block@arch.ethz.ch

(Submitted on 01/03/2012, Reception of revised paper 31/07/2012, Accepted on 31/08/2012)

ABSTRACT: This paper presents a new computational framework based on Thrust Network Analysis (TNA) for the design of funicular structures. Fast and robust solving algorithms enable the interactive exploration of these constrained structural systems. By giving explicit, bidirectional control over the internal force distribution and overall geometry to the designer, free exploration of these statically highly indeterminate systems is made possible. The equilibrium of funicular compression networks is represented by reciprocal diagrams, which visually express the force dependencies between different parts of the structure. By modifying these diagrams in real-time, the designer is able to explore novel and expressive vaulted geometries that are blurring the difference between shapes associated to typical compression-only forms, obtained e.g. with hanging networks, and freeform surface structures. The power of this framework for design is demonstrated by a user-friendly software implementation, which has been used to design and build a freeform, thin-tile masonry vault.

Key Words: Funicular form finding; Thrust Network Analysis; real-time structural design tools; interactive, bidirectional exploration; compression-only vaults; reciprocal diagrams

1. INTRODUCTION

In the last two decades the rise of computer-aided design and modelling techniques enabled a new language of doubly curved surfaces in architecture. The relation between architectural design and structural concepts has changed towards an integrated organizing principle of form, material and structure [1], which resulted in the emergence of a new design culture right at the border of architecture and engineering [2]. Through new digital fabrication methods, the realization of complex forms became furthermore technically and economically feasible. In order to achieve an efficient and elegant design for these non-standard structures, a close collaboration

between architects and engineers from early design phases, based on new, shared computational tools, gained importance [3]. To enable an intuitive approach to design problems with hard engineering constraints though, visual representation [4] and real-time feedback [5] of structural information is essential.

This paper discusses the problem of curved surface design within the rigorous framework of funicular, compression-only vault design. Historically, particularly hanging models and graphic statics have been used to design vaulted structures. In the beginning of the 20th century, Antoni Gaudí used hanging models in the design process of the Crypt of Colònia Güell [6]; Frei Otto and his team used hanging models to find the form

*Corresponding author e-mail: rippmann@arch.ethz.ch

for the lattice shell in Mannheim [7]; and Swiss engineer Heinz Isler designed his concrete shells based on hanging cloth models [8]. Around the same time as Gaudí, the Guastavino Company was designing large thin-tile vaults for important buildings all over the United States using graphic statics [9]. Such form-finding techniques, both physical and graphical, allow the exploration of three-dimensional systems, but the design process is time-consuming and tedious, particularly due to a lack of global control, since each local change affects the overall geometry.

In the last 15 years, a few three-dimensional computational methods have been developed for the equilibrium design of vaults. Kilian developed a virtual interactive hanging string modelling environment, with real-time exploration made possible through fast modelling and solving algorithms, adopted from cloth modelling used in the computer graphics industry [4, 10]. His approach emphasised the exploration experience, but had challenges to steer the design in a controlled manner. This was addressed by Andreu et al. [11], but applied only in the realm of equilibrium analysis of vaulted masonry structures, and without the interactive design component. Most recently, several interactive tools allowing for real time exploration of funicular networks have been developed, such as [12, 13]. Finally, Thrust Network Analysis (TNA), extending graphic statics to the third dimension for vertical loading, enables the explicit representation and control of all degrees of freedom in statically indeterminate funicular networks, by means of reciprocal diagrams and linear optimization [14, 15].

This paper describes an interactive, bidirectional design framework for compression-only vaults, based on the concepts of TNA. The new, underlying computational method that allows real-time feedback, is explained in detail. The paper is structured as follows: in Section 2, the basic concepts of graphic statics, reciprocal diagrams and TNA are explained; Section 3 describes the setup of the interactive design workflow, while Section 4 illustrates the developed solving methods in detail; Section 5 discusses questions of implementation, and presents an application of this framework, in the form of a prototypical shell; Section 6 focuses on future research; and the conclusions of this study are given in Section 7.

2. THEORETICAL BACKGROUND

This section describes the underlying concepts that form the foundation of the new design framework. These concepts, graphic statics and its three-dimensional extension, TNA, have been adapted and extended to enable real-time feedback.

2.1. Graphic statics and reciprocal diagrams

Graphic statics is a method for design and analysis of structures based on geometry and drafting, developed in the second half of the 19th century [16, 17]. It uses two diagrams: a *form diagram*, representing the geometry of the pin-jointed structure, and a *force diagram*, also referred to as *Cremona diagram* or *Maxwell-Cremona diagram*, representing the equilibrium of the internal and external forces of the structure (Fig. 1a-b). The power of graphic statics is based on its inherent bidirectional capabilities; one can either use the form diagram to construct the force diagram, or apply the inverse process and construct parts of the form diagram from an intended force diagram, i.e. either form or force constraints can drive the design exploration [5].

The relation between form and force diagram is called *reciprocal* [18], having the following topological, geometrical and structural properties:

- the form and force diagrams are *dual figures*, i.e. both diagrams have the same number of edges, and each node with a valency higher than one in

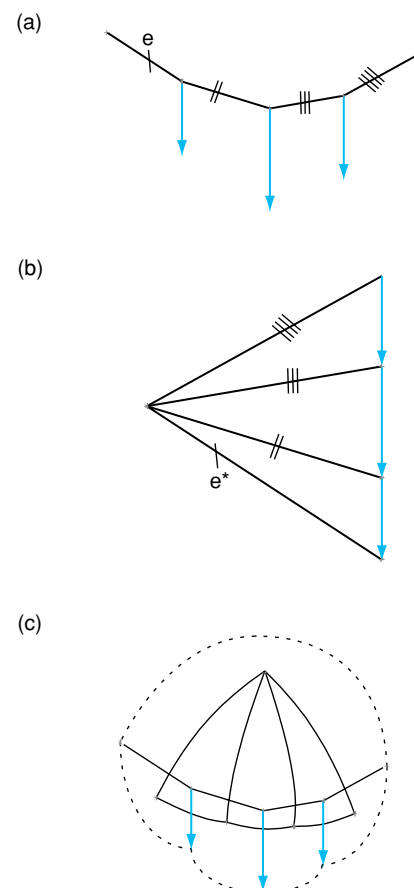


Figure 1. a) The form diagram for a tension/hanging funicular, b) its force diagram, and c) the dual relationship between both diagrams.

- one corresponds to a *space*, formed by a polygon of edges, in the other, and vice versa (Fig. 1c);
- each edge e in the form diagram (Fig. 1a) has a corresponding edge e^* , parallel to edge e , in the force diagram (Fig. 1b); and
- the length of edge e^* in the force diagram is, up to a scale factor, equal to the magnitude of axial force in edge e in the form diagram.

Note that for edges coming together at “internal” nodes of the form diagram, or in other words, their corresponding edges in the force diagram form closed polygons, representing closed vector polygons of forces. These reciprocal properties thus lead to a unique force diagram for a given statically determinate structure and loading case. If the structure is statically indeterminate, more than one reciprocal diagram exists for the given form diagram.

2.2. Thrust network analysis

Based on reciprocal diagrams and linear optimization, Thrust Network Analysis (TNA) provides a method for the design and analysis, for vertical loading, of vault geometries using discrete funicular networks (Fig. 2). These networks are not necessarily actual structures, but rather spatial representations of compression forces in equilibrium. Because of the vertical loading constraint, the equilibrium problem can be decomposed in two sequential steps:

- Step 1 - Solving horizontal equilibrium:* Since the vertical loads P vanish in Γ , which is defined as the horizontal projection of the thrust network G , the in-plane equilibrium of Γ also represents the horizontal equilibrium of G , independent of the

applied (vertical) loads (Fig. 2). Thus, this equilibrium state of Γ , represented by its reciprocal force diagram Γ^* and scale ζ , can be defined in a first, independent step.

- Step 2 - Solving vertical equilibrium:* For a given horizontal projection, Γ , and equilibrium of the horizontal force components, given by Γ^* and scale ζ (these have been defined in Step 1), a unique thrust network G , in equilibrium with the given loading P , is then found for each set of boundary vertices, V_F .

This two-step approach highlights the design parameters in the TNA framework, which allow to control the multiple degrees of freedom in statically indeterminate networks: a) the plan geometry and layout in plan of the forces of the structure, Γ ; b) the proportional distribution of (horizontal) forces, Γ^* ; c) the vertical rise of the equilibrium network G , inversely proportional to scale of Γ^* , ζ ; and, d) the support heights, V_F . Note that the control given by the scale factor ζ is analogous to moving the *pole* of a funicular polygon (along its *closing string*) in graphic statics [19]. Furthermore, the explicit control of the distribution of forces, by manipulation of the force diagram, allows to indirectly control the stiffness distribution of the thrust network during the form finding, independent of the choice of material, or in other words to explore the indeterminacy of statically indeterminate networks.

To guarantee compression-only solutions, it is necessary that all spaces of Γ , closing all unbounded spaces with an additional edge (dotted lines in Fig. 2), and Γ^* are convex polygons and have certain constraints on their orientations [15]. These additional constraints on both diagrams make this a hard problem. A key contribution of this paper is that it presents a new, robust algorithm for enforcing these constraints for large networks with any topology. The following section describes the integration of these solving procedures in the overall design process to allow an interactive, bidirectional and weighted modification of the two diagrams in real-time, and hence facilitate the intuitive design of the spatial thrust networks, all while enforcing the compression constraints.

3. DESIGN WORKFLOW

The TNA framework has been refined and expanded, introducing user-controlled, interactive modification possibilities of the form and force diagram in a bidirectional and weighted manner. The active control of the diagrams enables the user to steer the thrust

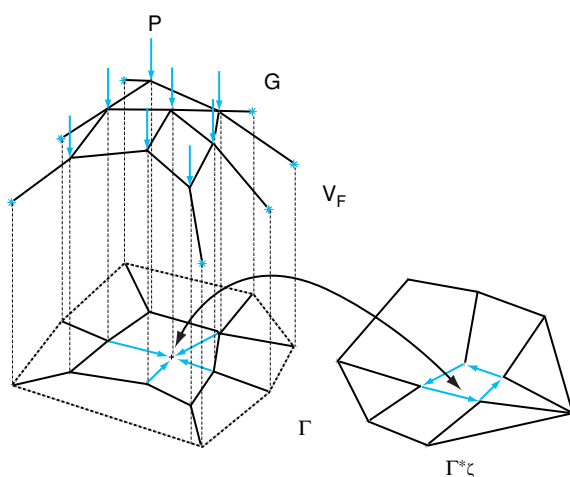


Figure 2. Thrust Network Analysis: form diagram Γ , force diagram Γ^* , with given scale ζ , the reciprocal relation between one node in the form diagram and corresponding space in the force diagram, and the thrust network G for given supports V_F and loading P .

networks interactively towards an intended shape. These new adaptations of the framework pave the way for a digital, TNA-based structural design tool, which is accessible and generally comprehensible to both architects and engineers.

During this digital design process, two sequential solving procedures, reflecting the two-step solving of the equilibrium described in Section 2.2, are repeated. The first procedure restores horizontal equilibrium, by enforcing corresponding edges of form and force diagram to be orientated in the same direction, which includes that they have to be parallel, while keeping the original connectivity for both diagrams. The second, following procedure then solves vertical equilibrium of the thrust network based on the obtained horizontal equilibrium in the first procedure, and for a given scale ζ .

The integration of both solving procedures in the overall design process is illustrated in the flow diagram in Figure 3. Details of the algorithms developed for both procedures are described in Section 4.

A typical design exploration has the following stages and components:

- The form diagram Γ is defined by the user, representing the force topology of the structure in plan (black lines in Fig. 3a), and can be generated or freely drawn.
- To generate a first reciprocal force diagram Γ^* , a dual diagram of Γ is generated by connecting the barycentres of its spaces (blue dotted lines in Fig. 3a). This diagram is then rotated counter-clockwise by 90 degrees, which provides a good starting point for the iterative procedure which enforces horizontal equilibrium, i.e. imposes the constraint that corresponding edges of Γ and Γ^* are parallel within a given tolerance (Fig. 3b).
- At this point, there is the possibility for the user to directly modify Γ or Γ^* , reflecting formal or structural design intents. The user can transform the diagrams by moving, scaling or rotating individual or multiple edges and nodes. In Figure 3c for example, the edges in Γ^* , corresponding to the highlighted line in Γ , are stretched, while keeping the other parts of Γ^* unaltered, which thus corresponds to proportionally attracting more force along that line.
- After the above-described modification, the modified diagram in Figure 3c no longer represents the equilibrium of Γ , because corresponding edges of Γ and Γ^* are no longer orientated in the same direction. Horizontal equilibrium thus needs to be enforced again (Fig. 3d).

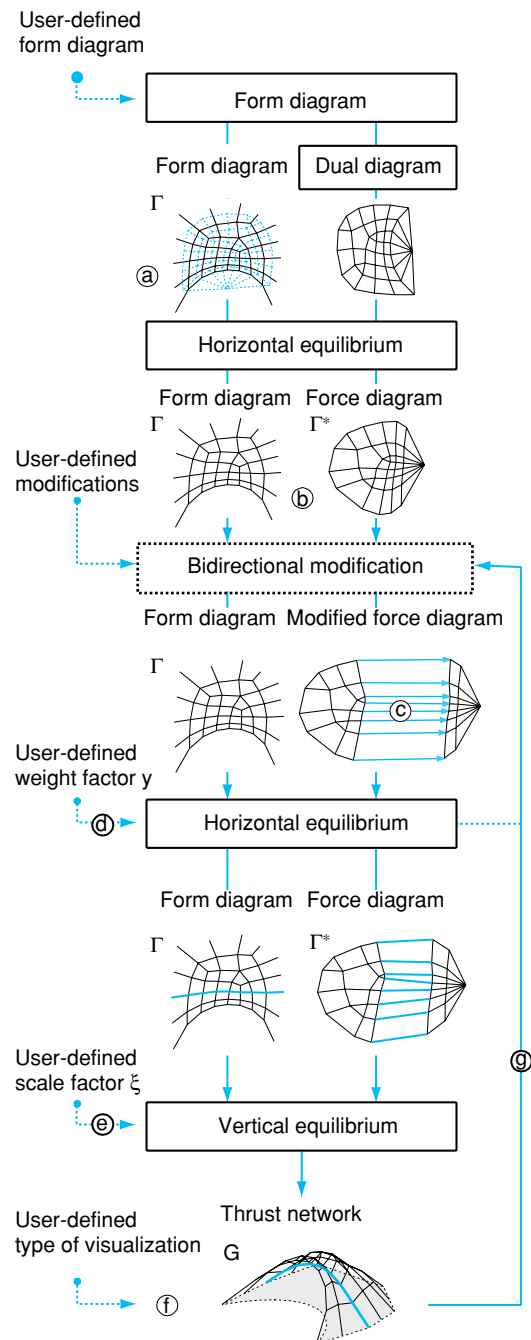


Figure 3. Overview of the main steps of the design workflow.

This procedure can be weighted to give priority to the geometry of either form or force diagram. Practically, this means that one can choose which diagram will be adapted less during the solving process that enforces parallelity. This will be discussed in more detail in Section 4.1.

- Based on the equilibrated Γ and Γ^* , a user-defined scale factor ζ , defining the overall rise of the solution, and given boundary heights, the nodal heights of the thrust network G are calculated to achieve vertical equilibrium for the loading applied at the nodes (Fig. 3e).

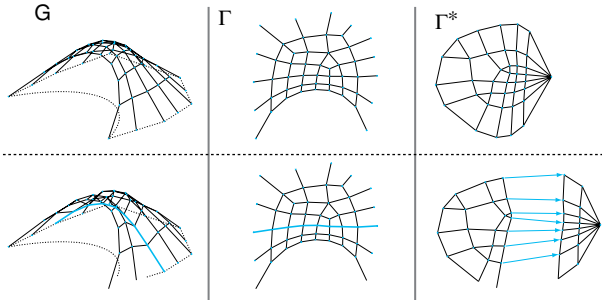


Figure 4. Resulting thrust networks G due to different user-controlled modifications of the force diagram Γ^* for a fixed form diagram Γ .

- The three-dimensional visualization of the thrust network G allows to check and evaluate the solution with respect to the design intent (Fig. 3f). Indeed, the manually enforced force redistribution, by the stretching of edges in the force diagram, has caused a fold, which can be understood as a structural rib, along the corresponding in-line edges in the thrust network. The user-controlled modification of the form and force diagram, followed by the solving procedures which in two steps enforce horizontal and vertical equilibrium, is repeated until the shape of the resulting thrust network meets the design requirements (Fig. 3g).

Figure 4 demonstrates for two stages in the design workflow the relationship of the thrust networks G to their corresponding form and force diagrams.

The following section describes the algorithms that have been developed to efficiently and robustly solve for the horizontal and vertical equilibrium of the reciprocal form and force diagrams, and the thrust network respectively.

4. SOLVING METHODS

The equilibrium of a typical node N_i of the form diagram Γ , with corresponding node V_i in the thrust network G and vertical load P_i applied to it, is shown in Figure 5. The horizontal equilibrium of node V_i , i.e. the in-plane equilibrium of node N_i , is represented by the three edges \mathbf{e}_{ij}^* , \mathbf{e}_{ik}^* , \mathbf{e}_{il}^* that form a closed polygon in the force diagram Γ^* . The length of these reciprocal edges, multiplied with the scale factor of Γ^* , ζ , is equal to the magnitude of the horizontal force components in the corresponding edges \mathbf{v}_{ij} , \mathbf{v}_{ik} , \mathbf{v}_{il} , respectively the axial forces in edges \mathbf{e}_{ij} , \mathbf{e}_{ik} , \mathbf{e}_{il} .

4.1. Horizontal equilibrium

The network in Figure 6 will be used to illustrate the iterative procedure that enforces horizontal equilibrium.

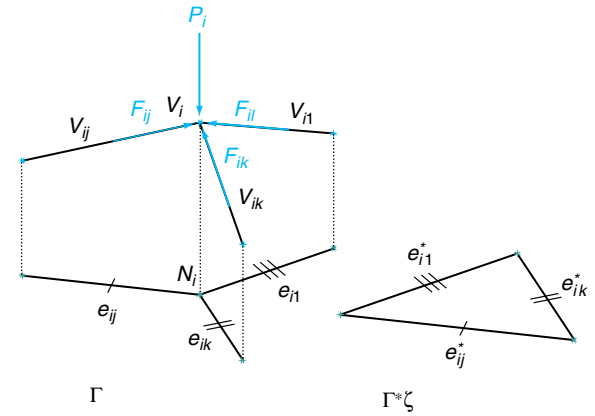


Figure 5. A typical node V_i of the thrust network G with adjacent edges \mathbf{v}_{ij} , \mathbf{v}_{ik} , \mathbf{v}_{il} , the axial forces F_{ij} , F_{ik} , F_{il} in these edges and the vertical load P_i . The projection of node V_i is N_i in Γ , with adjacent edges \mathbf{e}_{ij} , \mathbf{e}_{ik} , \mathbf{e}_{il} ; corresponding edges \mathbf{e}_{ij}^* , \mathbf{e}_{ik}^* , \mathbf{e}_{il}^* are forming a closed polygon, defining a space in Γ^* .

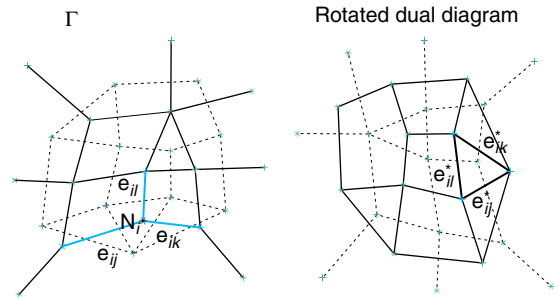


Figure 6. Three edges \mathbf{e}_{ij} , \mathbf{e}_{ik} , \mathbf{e}_{il} joined at node N_i in the form diagram Γ and their corresponding edges \mathbf{e}_{ij}^* , \mathbf{e}_{ik}^* , \mathbf{e}_{il}^* in the rotated dual diagram.

The image shows the form diagram Γ and its dual diagram (Section 3). The nodes of the form diagram Γ are defined by a set of nodes N . Note that the (rotated) dual diagram has as many spaces as nodes in Γ . For each diagram, the connectivity of the edges is defined by an adjacency list, associating each node with its neighbouring nodes which connect to it with edges. To find an allowed horizontal, compression-only equilibrium, the corresponding edges of both diagrams need to be parallel, which is in general not the case for the rotated dual diagram.

The solving procedure is described for a typical node N_i in Γ (Fig. 6). The edges \mathbf{e}_{ij} , \mathbf{e}_{ik} , \mathbf{e}_{il} , coming together in N_i , have the corresponding dual edges \mathbf{e}_{ij}^* , \mathbf{e}_{ik}^* , \mathbf{e}_{il}^* in Γ^* . These dual edges form a closed polygon, but are not parallel to their corresponding edges \mathbf{e}_{ij} , \mathbf{e}_{ik} , \mathbf{e}_{il} , which means that they do not describe the force equilibrium of node N_i in Γ . The degree of disequilibrium of node N_i is represented by the angle deviations between the corresponding edges in form and force diagram, highlighted in Figure 7.

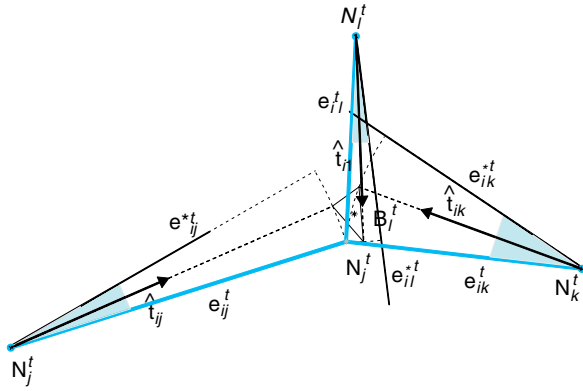


Figure 7. The normalised target vectors $\hat{\mathbf{t}}_{ij}^t, \hat{\mathbf{t}}_{ik}^t, \hat{\mathbf{t}}_{il}^t$, scaled by the length at step t of the corresponding edges $\mathbf{e}_{ij}^t, \mathbf{e}_{ik}^t, \mathbf{e}_{il}^t$, are added to the neighbouring points N_j^t, N_k^t, N_l^t to calculate the barycentre B_i^t .

Allowed geometrical configurations for both diagrams require that, for all corresponding edges in form and force diagram, their normalised edge vectors are equal:

$$\hat{\mathbf{e}}_{ij} = \hat{\mathbf{e}}_{ij}^* \quad (1)$$

Notice that the requirement in (1) not only demands that corresponding edges are parallel, but furthermore have the same orientation [15]. The horizontal equilibrium can furthermore be weighted by using the weighting factor $\gamma = \{0, \dots, 1\}$. This factor increases or decreases the influence of the form, respectively force, diagram on its counterpart during the iterative process. To achieve this, a target vector \mathbf{t}_{ij} for each pair of corresponding normalised edge vectors $\hat{\mathbf{e}}_{ij}$ and $\hat{\mathbf{e}}_{ij}^*$ is defined as follows:

$$\mathbf{t}_{ij} = \gamma \hat{\mathbf{e}}_{ij} + (1 - \gamma) \hat{\mathbf{e}}_{ij}^* \text{ with } 0 \leq \gamma \leq 1 \quad (2)$$

If $\gamma = 0$, only the edges of the form diagram are affected, respectively if $\gamma = 1$, only the edges of the force diagram are affected. This weighting factor thus allows to define to what degree each diagram stays fixed.

To find the equilibrium state of both diagrams for which corresponding edges have the same direction, an iterative solving procedure is used that updates the positions of N and N^* . This is done by iteratively changing the directions and lengths of all edges \mathbf{e} and \mathbf{e}^* . These are given by the target vectors, defined at the start of the iterative procedure, whereas the individual length of every edge is successively updated at each iteration step, based on the corresponding edge length defined at the previous iteration step. Since the target vectors are defined at the start of the iterative solving,

both diagrams can be processed separately until all corresponding edges have the same direction as the target vectors. It is thus sufficient to explain the iterative process by only considering, for example, the form diagram. This is what is done next.

As illustrated in Figures 7 and 8, the solving procedure for the horizontal equilibrium works as follows:

- Starting at step $t = 0$, at each step t and for each node N_i^t , the barycentre B_i^t is found by adding the corresponding, scaled target vectors $\hat{\mathbf{t}}_{ij} \|\mathbf{e}_{ij}^t\|$ to all n_i neighbouring nodes N_j^t (Fig. 7):

$$\mathbf{B}_i^t = \frac{\sum_j \left(N_j^t + \hat{\mathbf{t}}_{ij} \|\mathbf{e}_{ij}^t\| \right)}{n_i} \quad (3)$$

- The horizontal residual $\mathbf{R}_{h,i}^t$ at step t is defined as the vector between node N_i^t and barycentre B_i^t :

$$\mathbf{R}_{h,i}^t = \overline{N_i^t B_i^t} \quad (4)$$

The nodal position of N_i^{t+1} , at the next iteration $t + 1$, is then obtained as:

$$\mathbf{N}_i^{t+1} = \mathbf{N}_i^t + \mathbf{R}_{h,i}^t \quad (5)$$

This iterative approach is applied to all nodes until the stopping criteria are reached, i.e. corresponding edges have the same direction within a chosen maximum deviation angle, α . The angle difference between corresponding edges in form and force diagram is evaluated using the dot product of their normalised vectors with the target $\hat{\mathbf{t}}_{ij}$. For example, the evaluation of the edges of the form diagram is:

$$\left| \hat{\mathbf{e}}_{ij} \cdot \hat{\mathbf{t}}_{ij} \right| \leq \cos \alpha \quad (6)$$

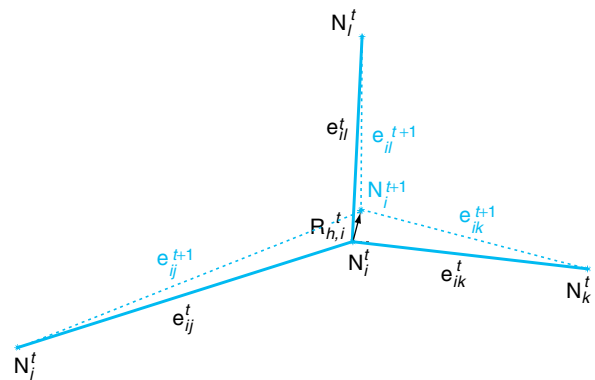


Figure 8. Node N_i^{t+1} at the next iteration is defined by node N_i^t and the horizontal residual $\mathbf{R}_{h,i}^t = \overline{N_i^t B_i^t}$ (see Fig. 7).

A maximum deviation angle α can be used to allow a limited angle tolerance between corresponding edges, causing minor inaccuracies in the thrust network but minimizing the number of iterations, respectively the calculation time (see Section 4.3). If the dot product in the LHS of (6) renders a negative value though, then these edges have flipped, i.e. changed orientation, during the iterative solving. These would correspond to local occurrences of tension forces in the structure. As pointed out in 2.2, indeed diagrams with non-overlapping, convex spaces are a necessary condition for compression-only thrust networks. The appearance of flipping edges during the iterative solving is typically observed when very small edge lengths occur in the diagrams. To prevent this, additionally, the length of every edge is constrained to be larger than a minimum length, defined proportionally with respect to the maximum lengths in the diagram.

Figure 9 visualizes the iterative procedure applied to both diagrams for a weighting factor $\gamma = 0.8$, showing the trace of the converging form and force diagram towards a horizontal equilibrium state. The chosen weighting factor puts less priority on the force diagram, thus making it adapt more strongly, steered by the edge directions of the form diagram. Oppositely, the directions of the edges in the form diagram only change marginally. As the boundary nodes of the form diagram represent the supports of the structure, they were constrained to remain fixed during the iterative process.

4.2. Vertical equilibrium

In the TNA approach, as described in Section 2.2, the equilibrium shape of G is obtained for a given form diagram Γ , force diagram Γ^* , scale ζ , loads P , and fixed supports V_F . In the case of a given set of loads P , the problem of finding the thrust network G is a linear problem [14, 15]. For vault design, this is only a valid assumption, when, for example, the thickness of the vault is adapted after the solving process such

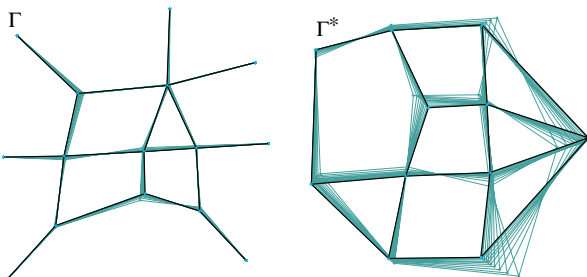


Figure 9. Visualisation of the iterative procedure, showing the form diagram Γ and force diagram Γ^* converging towards a state of horizontal equilibrium; weighting factor $\gamma = 0.8$.

that the nodal dead load is proportional to the assumed P . If rather a specific vault thickness, for example constant, is intended, then the loading P cannot be considered as given, as it is changing during the form-finding process. In this case, the loading thus needs to be related to the spatial tributary volume/area of each node of G , which means that the vertical equilibrium equations are no longer linear. Therefore, the solving for G is done through an iterative procedure, adapting the loading P to the tributary volume of each node after a certain number of steps during the solving process.

For the thrust network G , the equilibrium of a typical node V_i , with forces F_{ij} in the neighbouring edges v_{ij} (Fig. 5), can be described as:

$$0 = \sum_j F_{ij} + P_i \quad (7)$$

After running the solving procedure described in the Section 4.1, the horizontal force components in the edges v_{ij} , v_{ik} , v_{il} , defined by the lengths of the edges e_{ij}^* , e_{ik}^* , e_{il}^* in Γ^* and its scale ζ , are guaranteed to represent (within the chosen tolerance α) a compression-only horizontal equilibrium. This means that the force densities q_{ij} [20] of all edges are defined by Γ and Γ^* , and ζ [15]. For example for edge v_{ij} , one has:

$$F_{ij} = v_{ij} * q_{ij} = v_{ij} * \frac{|e_{ij}^{*t}|}{|e_{ij}^t|} * \zeta \quad (8)$$

The solving procedure for the vertical equilibrium works as follows:

- Start with a network G^0 , which has Γ as its horizontal projection, and goes through the supports V_F .
- Starting at step $t = 0$, at each step t , the residual force $R_{v,i}^t$ for each node V_i is computed as:

$$R_{v,i}^t = \sum_j F_{ij}^t + P_i^t \quad (9)$$

Note that the residual $R_{v,i}^t$ is vertical, since the horizontal force components of F_{ij}^t are in equilibrium (8), and P_i^t is vertical by definition.

- Using (8) and (9), the vertical residual $R_{v,i}^t$ is computed at each iteration step, and the nodal position is updated using a direct Euler method [21]:

$$V_i^{t+1} = V_i^t + R_{v,i}^t \quad (10)$$

The procedure converges as soon as $|R_{v,i}^t| < \varepsilon$ for all i , ε being a defined threshold value.

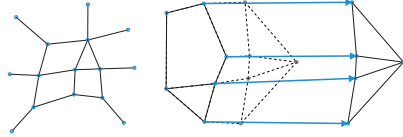
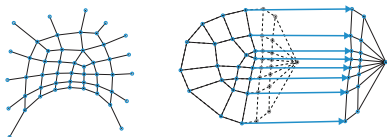
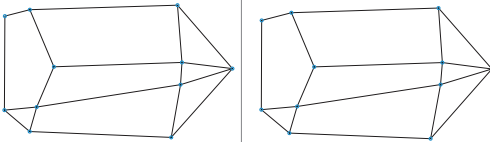
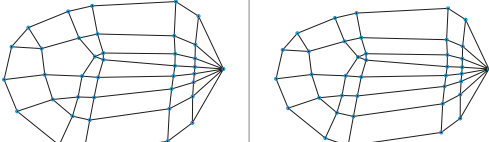
	Example A		Example B	
	Branches: 18 Nodes: 15 / 11		Branches: 74 Nodes: 49 / 39	
Form / Force Diagram Not in Equilibrium within given Angle Tolerance				
Angle Tolerance	$< 1_i$	$< 2_i$	$< 1_i$	$< 2_i$
Force Diagram In Equilibrium within given Angle Tolerance				
Euler Method	2.15 ms (111)	1.17 ms (54)	472.77 ms (5732)	195.56 ms (2042)
Runge-Kutta Method (fourth order)	2.11 ms (49)	1.72 ms (35)	131.06 ms (781)	22.11 ms (135)

Figure 10. Calculation times and iterations (in parenthesis) for two example diagrams, using different solvers and angle tolerances; solved on a standard PC (Intel® Core™, i7 CPU Q 820, 1.73 GHz).

4.3. Extensions and adaptations

To use the algorithms described above as interactive form finding tool, *Dynamic Relaxation* and *Runge-Kutta* solvers have implemented to reduce calculation time, while maintaining numerical stability of the methods [21] (Fig. 10).

A stable and flexible use of the form finding tool was addressed by allowing the control of specific parameters during the solving procedures. For example, nodes of the form and force diagram can be locked or assigned with a weighting factor by the user

to precisely control the position changes during the iterative process. This allows one for example to fix boundary nodes in the form diagram to maintain support locations or to exclude defined parts of the diagrams from global changes during the iterative geometric calculation process of the horizontal equilibrium.

For the approach presented in this paper, the example in Figure 11 demonstrates the capability of these methods for dealing with complex form and force diagrams, without losing interactivity. The example has a total number of 2084 edges and 1124 nodes that had to be solve for the horizontal (for each diagram) and vertical equilibrium respectively.

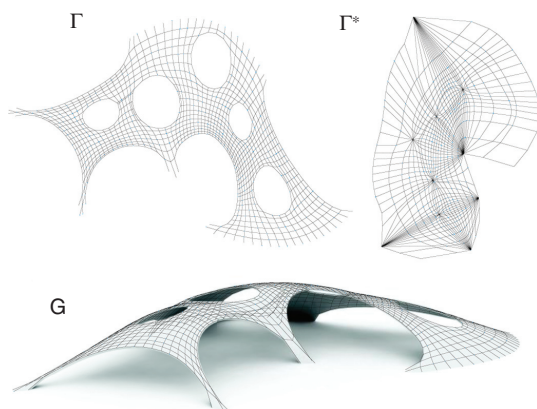


Figure 11. A more complex example with a total number of 2084 edges.

5. APPLICATION

This section describes the interactive form finding tool, RhinoVAULT [22], which has implemented the methods discussed in this paper. It highlights key features of the software implementation, developed to address practical challenges throughout the form finding process, and describes the fundamental ideas concerning the operation of the software. The structural design tool was used for the digital design process of a freeform timber vault [23], demonstrating its usability for real-world applications.

5.1. Digital structural design tool

The inherent, bidirectional interdependency of form and forces, represented in visual diagrams, is key for a user-driven and controlled exploration in the structural form finding process. Thus, the implementation and design of the form finding tool focused on a design approach through exploration, underlining the visual and playful nature of the approach. Its use is mainly targeting the early structural design stages allowing the exploration of structural form driven by the user's initial design intent within the constraints of compression-only equilibrium.

Structural design is often based on successively modifying the shape of a structure and using structural analysis to assess its structural behaviour in a second, independent step. This process can be automated, for example by using evolutionary structural optimization, to provide a more interactive design approach for the user [24]. RhinoVAULT, in contrast to such sequential approaches, emphasises the inherent simplicity and visibility of the graphical approach to explicitly steer form and forces. This not only fosters the understanding of the form finding process, but promotes knowledge of structural design in general. This educational value of the developed tool has been confirmed in the course and outcome of several form finding workshops organized for architects and structural engineers.

RhinoVAULT was developed as a plugin for the CAD modelling software Rhinoceros® [25] using RhinoScript [26], a procedural scripting language based on Visual Basic Script, and C#.

5.2. Form finding with RhinoVAULT

Building on the experience and user feedback gained throughout several workshops, the plugin's structure and functionality has been tested with respect to its purpose for design. The use of the plugin is based on the inherent procedural process of the TNA framework, embedded in the design workflow as described in Section 3. It is itemised by basic steps, which are controlled by the user during the design exploration.

Figure 12 specifically addresses the topological characteristics of the form and force diagrams. Configuration I shows the initial form diagram, representing a possible layout of forces of the thrust network in plan. In configuration II, an open, unsupported edge arch is introduced, resulting in a funicular edge in the form diagram (top left) and its typical fan-shaped force polygon in the force diagram (right). The equilibrium of the inserted opening in the

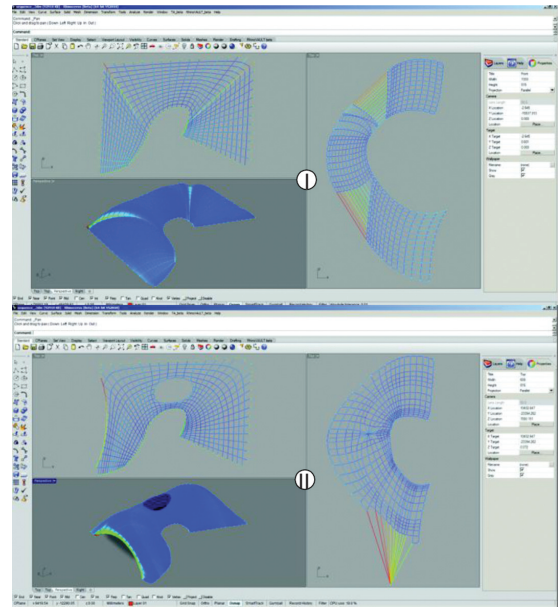


Figure 12. Screenshots of RhinoVAULT in Rhinoceros®, highlighting the topology change of the form diagram by introducing a non-supported edge and an opening and after relaxing the diagrams.

middle of the vault, obtained by deleting some edges of configuration I, is represented by the star-shaped part in the force diagram. Furthermore, a relaxation of the form diagram was applied to obtain configuration II, which also explains its smoother force distribution in comparison to configuration I, without the localised force concentrations.

Figure 13 shows the design process of the built vaulted structure shown in Figure 14. With the

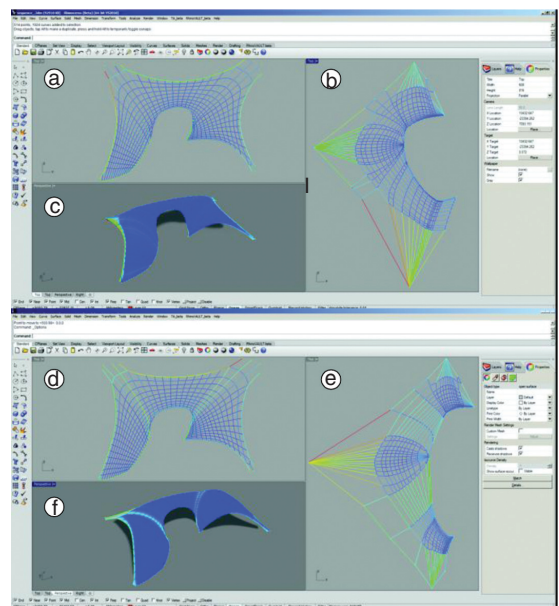


Figure 13. Screenshots of RhinoVAULT in Rhinoceros®, showing for two similar form diagrams (a, d), two (b, e) possible force diagrams and the resulting thrust networks (c, f).



Figure 14. Built prototype of a freeform, unreinforced timber vault at ETH Zurich, Switzerland, 2011, based on the structural design process shown in this paper.

generation of the initial form diagram (Fig. 13a), the user defines the vault in plan and its support conditions, respectively the number of unsupported edge arches of the thrust network (Fig. 13c). Consequently, one identifies five fan-shaped funicular polygons in the force diagram, representing the funicular equilibrium in plan of these unsupported edge arches (Fig. 13b). Comparing both form diagrams (Figs. 13a and 13d), one identifies that they are very similar. The individual edge lengths of the force diagram of the second (Fig. 13e) have been locally modified by the user though, resulting in a more differentiated force distribution. Such an operation is possible due to the indeterminacy of the force diagram. For example, stretched parts of the force diagram attract more force along two continuous force paths in the form diagram, causing two folds in the thrust network (Fig. 13f). Furthermore, the scaled-down bottom part of the force diagram globally reduces the values of the horizontal thrusts in that part, which thus results in a higher rise of the left section of the thrust network. These examples of possible modifications of the force diagram, re-distributing the forces within the structure, demonstrate an effective and interactive way to control the form finding process and shape the resulting thrust surface. Thanks to the iterative solving procedure, which enforces horizontal equilibrium in a weighted manner after each modification, these can furthermore be applied in an exploratory and free manner.

The roughly 30 square metre freeform, unreinforced timber vault shown in Figure 14 has been realised with RhinoVAULT. The local thickness of the vault was directly informed by the graphical information provided by the form and force diagram, putting more material only where it was locally needed. Note that the longest edge lengths in the force diagram are the areas with highest horizontal forces (Fig. 13d).

Besides the geometrical information, e.g. the lengths of edges in the force diagram, colours help to identify

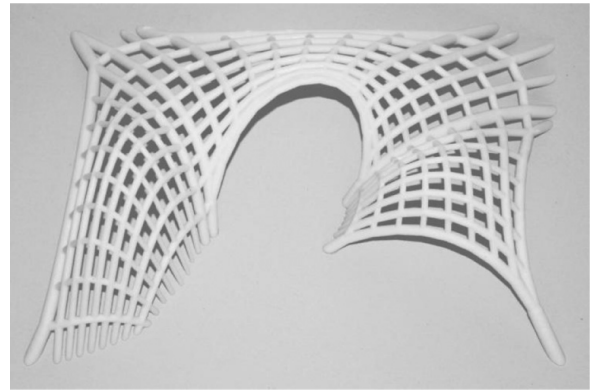


Figure 15. A three-dimensional pipe diagram (physical model) showing the non-uniform distribution of forces in the freeform vault shown in Figs. 13d-f and 14.

areas with maximum or minimum compression force. In addition, pipe diagrams are used to visualise the three-dimensional force distribution in the thrust network. Varying radii, based on the local force magnitudes, indicate the compression forces in the elements (Fig. 15). For a better three-dimensional representation, a faceted mesh forms a continuous thrust surface (Fig. 13c, f). These features help to read the diagrams, identify corresponding edges and localise force concentrations.

The tool further incorporated several features that increase the level of control over the form finding process. As most powerful application, geometrically constraining the force diagram can be used directly to optimise specific structural criteria. For example, edges can be constrained to a user-defined maximum and minimum length, thus limiting the thrust magnitudes within the structure to a distinct range. By varying the “inertia” of individual or groups of nodes during the solving for horizontal equilibrium, it is furthermore possible to control this behaviour locally.

6. DISCUSSION AND FUTURE WORK

Besides the presented intuitive exploration of structural form, designers often want to achieve a specific, desired form, especially if the initial design idea was developed focusing on formal or spatial rather than structural considerations. By experience, one can steer the shape towards the intended goal by gradually modifying the force diagram in a back-and-forth process. An alternative approach is to embed the forward TNA method into a parameter search, which controls all degrees of freedom to find a best-fit compression solution to a given target surface [27]. Considering the techniques described in this paper, an obvious next step is to adapt the idea of best-fit compression solutions to the presented methods, with

as goal to allow for “tri-directional” control, so not only exploring structural form via the form and force diagrams, but also directly using a three-dimensional target geometry.

The initial topology of the form diagram significantly influences the achievable, possible equilibrium shapes [5, 15]. Research on the generation of starting form diagrams in relation to an intent, as provided e.g. by a three-dimensional target surface representing the vault, will be part of future work. Also, subdivision schemes to change the density of the diagrams and hence the accuracy of the results wants to be explored further.

It is important that one bears in mind that the form finding process is based on vertical loading, particularly appropriate for heavy masonry or concrete shells, but arguably also an acceptable assumption in the (early) design stages of shells of sufficient span, providing interesting starting points for further analysis and optimization processes. An analysis stage for asymmetric loading and buckling stays of course absolutely essential before realization, especially for lighter structures.

7. CONCLUSION

This paper introduced a new computational framework based on Thrust Network Analysis (TNA) for the design of funicular vaulted structures. The presented methods allow a fast and robust solving for the interactive exploration of these constrained structural systems.

The main contribution is the explicit, bidirectional control of the internal force distribution and overall geometry, to freely explore these statically highly indeterminate systems, without losing the transparency and comprehensibility of the process.

It was shown that the resulting methods for the horizontal and vertical equilibrium can be implemented in a modern CAD environment, to create a form finding tool that provides intuitive control over the complex relation between form and forces of three-dimensional compression-only structures.

The paper rendered the clear and understandable structure of the methods to allow for reproducibility, but also for extensions and adaptations to be integrated easily, underlining the flexible use of the framework for future developments.

The potential of the form finding process was demonstrated in the design and realisation of a freeform unreinforced timber vault. Due to the negligible or zero bending capacity of masonry, the realised prototypes [22] and other experimental

models [28] demonstrate the architectural possibilities of compression-only structures, furthermore validating the approach presented in this paper.

REFERENCES

- [1] Oxman, R., Morphogenesis in the Theory and Methodology of Digital Tectonics, *Journal of the International Association For Shell And Spatial Structures*, 2010, 51(3), 195–205.
- [2] Oxman, R. and Oxman, R., Eds., *The New Structuralism: Design, Engineering and Architectural Technologies*, John Wiley, New York, 2010.
- [3] Tessmann, O., *Collaborative Design Procedures for Architects and Engineers*, PhD thesis, University of Kassel, 2008.
- [4] Fergusson, E. S., The Mind’s Eye: Nonverbal Thought in Technology, *Science*, 1977, 26, 827–836.
- [5] Kilian, A., *Design exploration through bidirectional modeling of constraints*, PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 2006.
- [6] Tomlow, J., Graefe, R., Otto, F., and Szeemann, H., *The Model*, Institute for Lightweight Structures (IL), 34, University of Stuttgart, 1989.
- [7] Burkhardt, B., and Bächer, M., *Multihalle Mannheim*, Institute for Lightweight Structures (IL), 13, University of Stuttgart, 1978.
- [8] Chilton, J., *The Engineer’s Contribution to Contemporary Architecture: Heinz Isler*, Thomas Telford Press, London, 2000.
- [9] Ochsendorf, J., *Guastavino Vaulting – The Art of Structural Tile*, Princeton Architectural Press, New York, 2010.
- [10] Kilian, A., and Ochsendorf, J., Particle-Spring Systems for Structural Form Finding, *Journal of the International Association For Shell And Spatial Structures*, 2005, 46(2), 77–85.
- [11] Andreu, A., Gil, L., and Roca, P., Computational Analysis of Masonry Structures with a Funicular Model, *Journal of Engineering Mechanics*, 2007, 133(4), 473–480.
- [12] Piker, D., Kangaroo - Live 3-D Physics for Rhino/Grasshopper, computer software, 2011. <http://spacesymmetrystructure.wordpress.com/2010/01/21/kangaroo/>
- [13] Harding, J., and Shepherd, P., Structural Form Finding using Zero-Length Springs with Dynamic Mass, in: *Proceedings of the IABSE-IASS Symposium 2011*, London, 2011.
- [14] Block, P. and Ochsendorf, J., Thrust Network Analysis: A new methodology for three-dimensional equilibrium, *Journal of the International Association for Shell and Spatial Structures*, 2007, 48(3), 167–173.
- [15] Block, P., *Thrust Network Analysis: Exploring Three-dimensional Equilibrium*, PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 2009.
- [16] Culmann, C., *Die graphische Statik*, Meyer & Zeller, Zurich, 1864.
- [17] Cremona, L., *Graphical Statics: Two Treatises on the Graphical Calculus and Reciprocal Figures in Graphic Statics*, English Translation by Thomas Hudson Beare, Clarendon Press, Oxford, 1890.
- [18] Maxwell, J. C., On Reciprocal Figures and Diagrams of Forces, *Philosophical Magazine*, 1864, 4(27), 250–261.

- [19] Van Mele, T., Rippmann, M., Lachauer L., and Block, P., Geometry-based Understanding of Structures, *Journal of the International Association for Shell and Spatial Structures*, 2012, in press.
- [20] Schek, H.-J., The force density method for form finding and computation of general networks, *Computer Methods in Applied Mechanics and Engineering*, 1974, 3(1), 115–134.
- [21] Veenendaal, D., and Block, P., An overview and comparison of structural form finding methods for general networks, *International Journal of Solids and Structures*, 2012, in press.
- [22] Rippmann, M., Lachauer, L., and Block, P., RhinoVAULT - Designing funicular form with Rhino, computer software, 2012. <http://block.arch.ethz.ch/tools/rhinovault/>
- [23] Davis, L., Rippmann, M., Pawlofsky, T., and Block, P., Innovative Funicular Tile Vaulting: A prototype in Switzerland, *The Structural Engineer*, 2012, in press.
- [24] Preisinger, C., Diles, J., and Vierlinger, R., Karamba, Parametric Structural Modeling, computer software, 2012. <http://twl.uni-ak.ac.at/karamba/>
- [25] McNeel, R., Rhinoceros: NURBS modeling for Windows, computer software, 2011. <http://www.rhino3d.com/>
- [26] Rutten, D., *RhinoScript 101 for Rhinoceros 4.0*, Robert McNeel Associates, 2007.
- [27] Block, P., and Lachauer, L., Closest-Fit, Compression-Only Solutions for Free Form Shells, in: *Proceedings of the IABSE-IASS Symposium 2011*, London, 2011.
- [28] Block, P., Rippmann, M., and Lachauer, L., Validating Thrust Network Analysis using 3D-printed, Structural Models, in: *Proceedings of International Association for Shell and Spatial Structures Symposium 2010*, Shanghai, China, 2010.