

Architectural structures from quad meshes with planar parameter lines

Cheng Wang^a, Caigui Jiang^{a,b}, Hui Wang^a, Xavier Tellier^c, Helmut Pottmann^{a,*}

^aKing Abdullah University of Science and Technology, P.O. Box 2187, 4700 Thuwal, 23955-6900, Kingdom of Saudi Arabia

^bXi'an Jiaotong University, No.28 Xianning West Road, Xi'an, 710049, China

^cNavier Laboratory (Ecole des Ponts, UGE, CNRS), 6 av. Blaise Pascal, 77420, Champs-sur-Marne, France

Abstract

We address the computational design of architectural structures which are based on a frame of intersecting beams that are aligned with the parameter lines of a quad mesh. While previous work mainly put a planarity constraint onto the faces of the mesh, we focus on the planarity of long-range supporting beams which follow selected polylines in the underlying mesh. In addition to that, we impose further constraints including planarity of faces, right node angles and static equilibrium, and discuss in which way these may be combined. Some of the studied meshes are discrete counterparts of certain well-known surfaces in classical geometry, whose knowledge is helpful for initializing the proposed optimization algorithms.

Keywords: architectural geometry, computational design, computational fabrication, planar parameter lines, quad mesh, planar quad mesh, support structure, gridshell, funicularity, static equilibrium

1. Introduction

The design and fabrication of freeform shapes in contemporary architecture is a rich source for challenging problems in geometric computing [25]. An important type of structure possesses a quad mesh with regular combinatorics and possibly isolated combinatorial singularities as a basic underlying geometric model. The edges of the mesh are aligned with supporting beams and the faces correspond to panels. The cost of manufacturing panels and beams is an important factor in the fabrication of such a structure. A lot of attention has so far been payed to simple panels, in particular to flat ones. Much less do we know about the design of structures from planar long-range supporting beams which follow the dominant mesh polylines (parameter lines) in the quad mesh. Geometrically, this amounts to the design of quad meshes where one or both families of parameter lines are planar. Such structures are discrete versions of smooth surfaces with one or two families of planar iso-parameter lines. Planarity of long-range elements has a global influence on the overall shape, which becomes especially tricky if one pairs this with additional constraints such as planarity of faces, static equilibrium or right node angles. The present paper contributes to exactly these requirements, but does not study structures with two families of planar polylines and planar panels, since those have recently been carefully studied [13].

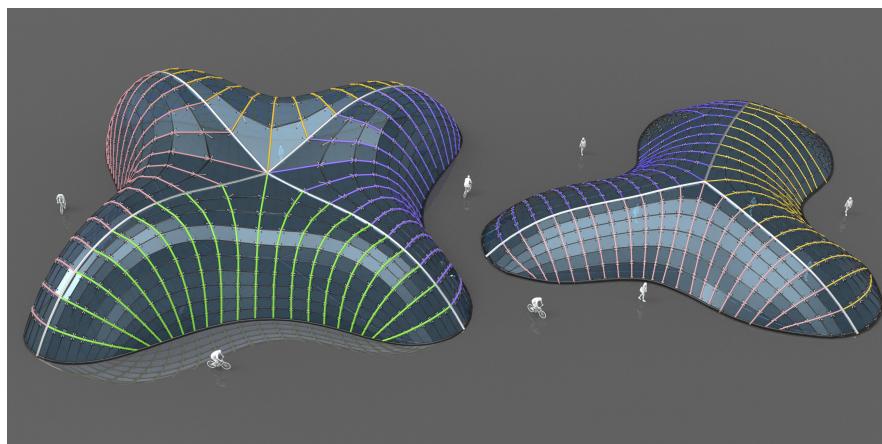


Figure 1: Left: A quad mesh with planar faces and one family of piecewise planar polylines (in colors) approximating the surface on the top of the Lilium Tower model by Zaha Hadid Architects. The detail of this design is discussed in subsection 2.2 and Fig. 6. Right: A quad mesh in static equilibrium rationalized with planar faces and one family of piecewise planar polylines (in colors). The detail of this design is discussed in subsection 3.2 and Fig. 10.

*Corresponding author

Email addresses: cheng.wang@kaust.edu.sa (Cheng Wang), caigui.j@gmail.com (Caigui Jiang), hui.wang.1@kaust.edu.sa (Hui Wang), xavier.tellier@enpc.fr (Xavier Tellier), helmut.pottmann@kaust.edu.sa (Helmut Pottmann)

14 *1.1. Planar beams in architectural structures*

15 *1.1.1. Beam continuity*

16 There are three possible ways to connect beams in a grid structure:

- 17 Beams are continuous throughout nodes, such as in the Schubert club band shell [17] (Fig. 2 (a)), where one family of
beams is stacked on top of the other.
- 18 Beams are interrupted at nodes, such as in the gridshell of the Joe and Rika Mansueto library [37] (Fig. 2 (b)), where the
straight beams are connected at nodes with curvature concentration.
- 19 In hybrid cases, beams are continuous in only one direction, and discontinuous in the others. This is for example the
case for the timber structures of the Cité du Vin in Bordeaux (Fig. 2 (c)) and the Seville Parasol [33], and for the steel
structure of the Sage Gateshead music center [34].

24 In gridshells, beam continuity shows advantages in simplifying connections and making them stiffer. It is also studied that
25 a timber gridshell with continuous beams can significantly reduce the environmental footprint [15]. This paper will focus on
26 structures with continuous beams, including cases 1 and 3. In these cases, planarity of the continuous beams yields remarkable
27 advantages in both beam fabrication and structural performance.



Figure 2: Planar beams in architectural structures. (a) Schubert Club Band Shell, designed by James Carpenter Design Associates (image from [flickr.com](#) photographed by Tony Webster under CC BY-SA 2.0). (b) Joe and Rika Mansueto Library, designed by Murphy/Jahn in 2011 (image from [wikimedia.org](#) photographed by Michael Barera under CC BY-SA 4.0). (c) La Cité du Vin in Bordeaux, designed by Anouk Legendre and Nicholas Desmazières XTU Agency in 2016 (image from [flickr.com](#) photographed by Adrien Sifre under CC BY-NC-ND 2.0).

28 *1.1.2. Advantages of planar beams*

29 *Fabrication.* Beams are usually fabricated from timber or steel. A common method to fabricate timber beams is gluing
30 laminated timber. Such beams are produced by gluing wooden strips which are bent in a plane. If a beam is not planar, each
31 strip needs to be further subdivided into rods (Fig. 3 (a) and (b)) so that manufacturing complexity increases drastically. Steel
32 beams are commonly produced by a roller bender from initially straight beams (Fig. 3 (d)). In this process, it is difficult to
33 obtain non-planar beams with sufficient precision since the three rollers should be aligned in a plane. Therefore for both timber
34 and steel, the fabrication of long-range planar beams is much cheaper and more effective.

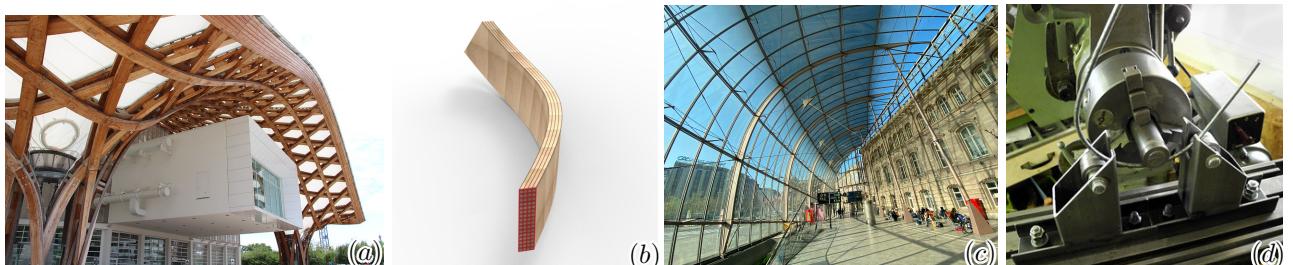


Figure 3: (a) Non-planar timber beams, such as the ones of the Centre Pompidou Metz (designed by Shigeru Ban), are curved in two directions (image from [flickr.com](#) photographed by Fred Romero under CC BY 2.0). (b) Hence, when fabricated by gluing laminated timber, the wooden strips must be subdivided into an array of rods (highlighted in red). (c) In the gridshell structure of the Strasbourg train station (designed by AREP), the planar beams are made from steel and can easily be reinforced by “bike-wheel” cable diaphragms. (d) A roller bender bends the steel with three rollers in a plane (image from [flickr.com](#) photographed by Fred Romero under CC BY-ND 2.0).

35 *Structural performance.* The planarity of beams also offers structural benefits. A straightforward advantage is that beam
36 planarity in both directions yields a torsion-free support structure which can simplify the node manufacturing [28]. It can also
37 facilitate the construction of multi-layer gridshells, such as in the Chinese National Center of Performing Arts. In addition,
38 planar continuous beams provide convenience for the installation of structural diaphragms. Cable diaphragms are for example

39 used in the Strasbourg train station to make beams slender and thus obtain a remarkably light structure [2] (Fig. 3 (c)).
40 Another example is the cable reinforcement on the glass roof of Hamburg history museum [32]. The design of such diaphragms
41 based on conics was recently studied in [41].

42 1.1.3. Compatibility of planar beams with further constraints

43 Even though beam planarity is a strong global constraint, there is sufficient freedom to combine beam planarity with other
44 constructive properties. A desirable feature in architectural design is planar quad panels. These are easily achieved with
45 translational shells, as seen at the Joe and Rika Mansueto library [37] (Fig. 2 (b)), and the Hippo House in the Berlin zoo [34].
46 Moreover, all quad meshes with planar faces and planar parameter lines possess a relation to translational nets, expressed in
47 terms of projective geometry [13].

48 Another architecturally interesting property is the so-called *funicularity* (or *static equilibrium*), which means that a surface-
49 like structure can resist the load with pure axial forces and has no bending moment. In a self-supporting structure all these
50 axial forces are compressive [43]. Principal meshes in static equilibrium have been studied in [23]. A grid may also admit a
51 self-stressing mode: the internal axial forces that are at equilibrium *without* external load. Such shapes may be realized as a
52 cable-net, a remarkably light-weight structural typology.

53 For architectural construction, it is also useful to have an orthogonal crossing angle of beams at the nodes. This generates
54 repetitive nodes in the whole structure and simplifies the node manufacturing process.

55 Planar beams can also be further constrained by considering the plane orientation. Architectural structures can benefit
56 from a family of planar supporting beams which are aligned in horizontal, i.e. parallel planes, for example following floor slabs.
57 Also, a gridshell with vertical supporting beams can offer a clear top view for a designer and even provide an aesthetic shading
58 pattern, such as in the Dutch Maritime museum [1].

59 The motivation of this paper is to identify those constraint sets which allow one to represent arbitrary freeform shapes,
60 and separate them from those which yield a restricted class of surfaces. The latter case is more frequent, but in view of
61 applications in architecture, it is not an obstacle. In fact, our constraints come from practical considerations in connection with
62 the architectural application.

63 1.2. Contributions and overview

64 In this paper, we present approaches to the computational design of quad meshes with one or two families of planar parameter
65 lines with additional properties which play a role in the architectural applications. Specifically, our contributions are as follows:

- 66 • In Section 2, we investigate surface approximation by quad meshes with planar parameter lines and planar faces. Since two
67 families of planar parameter lines and face planarity lead to a shape restriction [13], we provide a method for approximating
68 a given surface by a quad mesh with planar faces and one family of planar parameter lines. We also address cases where a
69 global solution does not exist, but one can combine quad meshes with the required properties to a useful global structure
70 with piecewise planar parameter lines.
- 71 • In Section 3, we study meshes with planar polylines and funicularity (static equilibrium), with and without considering
72 vertical loads. The former case is interesting from a geometric perspective (Subsection 3.1), the latter is more important
73 for practical design (Subsection 3.2). Using the so-called Airy stress surface, this requires only a small modification of
74 the algorithm presented in Section 2.
- 75 • In Section 4, we introduce two methods for finding meshes with planar polylines which meet at an angle of 90°. One
76 approach uses a special class of surfaces with planar orthogonal parameter lines for accessing the design space. The other
77 one starts with a numerical solution of the partial differential equation that needs to be solved in the case of smooth
78 surfaces. In both cases, a global optimization algorithm detailed in Section 5, can be used to modify the initial shapes
79 with mesh editing operations.
- 80 • Using the optimization approach described in Section 5, we refine the initialized structures and explore the design space
81 by shape editing. We further provide a number of designs illustrating the capabilities of the presented computational
82 design framework.

83 The constraints studied in this paper and their respective abbreviations are summarized in Table 1.

84 1.3. Previous work

85 *Related work on design for fabrication.* Meshes with planar polylines have been used in architectural construction and stylized
86 fabrication. Within architectural geometry, we point to work on meshes with planar polylines and planar faces, partially
87 also with a right node angle, based on generation methods which have their origin in classical and constructive geometry
88 [20, 39, 40]. Since a right node angle and planar faces characterize a discrete curvature line parameterization, one has a
89 relation to the classical subject of surfaces with one or both families of planar principal curvature lines. A Laguerre geometric
90 perspective and construction methods based on that have recently been proposed for computational design [13]. We refer to
91 this paper also for a review of the related classical geometric literature.

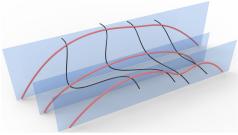
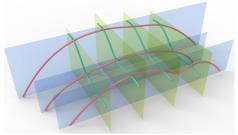
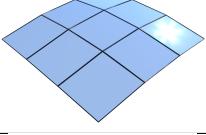
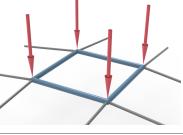
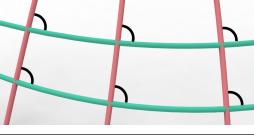
Abbreviations	Constraints	Figures
P	planar beams in one family	
PP	planar beams in both families	
PQ	planar quad faces	
F	funicularity (static equilibrium) with a vertical load	
90°	orthogonal crossing beams (right node angle)	

Table 1: Overview of constraints studied in this paper.

92 In addition, we mention research on triangle meshes of regular combinatorics where some or all of the dominant mesh
93 polylines are planar [9].

94 Besides the constructional advantage for architecture, shapes with planar sections also frequently appear in small- or
95 medium-scale stylized fabrication, for example in artworks called slice-forms [36]. McCrae et al. [18] presented an approach to
96 coarse shape approximation with just a few well-chosen planar slices, while the interlocked assembly of structures from planar
97 pieces is addressed by Schwartzburg and Pauly [35]. For more references on stylized fabrication we refer to the survey by Bickel
98 et al. [3].

99 *Related work on smooth surfaces* will be described in connection with the respective discrete versions in Sections 2, 3 and 4.

100 2. Combining planar polylines and planar faces

101 For many architectural applications, the skin of the structures can be treated as a quad mesh. The edges of the mesh will
102 be part of the support structure and the faces will be panels. There, it is a great advantage if the panels are planar. The
103 resulting planar quad meshes have therefore received a lot of interest in Architectural Geometry (see e.g. [14, 19, 28, 25]).

104 In this section, we study planar quad (PQ) meshes in which one family of polylines is planar (P+PQ meshes). The generation
105 of PQ meshes with both families of parameter lines being planar (PP+PQ meshes) is restricted in the possible shapes, for which
106 the form-finding methods have been studied in [13].

107 2.1. Smooth Surfaces

108 2.1.1. Planar parameter lines

109 Meshes with planar parameter lines are discrete versions of parameterized surfaces $\mathbf{x}(u, v)$ whose isoparameter lines $u = \text{const}$
110 and/or $v = \text{const}$ are planar. In this section, we will alternate between the smooth setting and its discrete versions. For some
111 cases, knowledge on the smooth counterparts is important to get insight on the feasibility of certain combinations of constraints.
112 It is also important for an appropriate initialization of the employed optimization algorithms.

113 In the following, M denotes a quad mesh with grid combinatorics. We consider it parameterized over a rectangular portion
114 of the \mathbb{Z}^2 lattice such that vertex $\mathbf{v}_{i,j}$ is the image of the point $(u, v) = (i, j) \in \mathbb{Z}^2$. The discrete parameter lines belong to
115 constant i or j in the parameter lattice. We are interested in those meshes M for which at least one family of parameter lines
116 is planar. If one or both families of parameter lines are planar, we call the mesh M a *P-mesh* or a *PP-mesh*, respectively.

117 Our meshes M may be seen as discrete counterparts to smooth surfaces S obtained by a parameterization $\mathbf{x} : \mathbb{R}^2 \supseteq$
118 $[u_0, u_1] \times [v_0, v_1] \rightarrow \mathbb{R}^3$, whose u -lines ($v = \text{const}$) or/and v -lines ($u = \text{const}$) are planar. The planes in which the u -lines lie

119 are denoted by $U(v)$, and likewise planar v -lines lie in planes $V(u)$. We use the same notation for meshes, but write $U_j = U(j)$
 120 and $V_i = V(i)$ for the planes of parameter lines.

121 Any surface S can at least locally be parameterized with planar parameter lines. We even have a lot of freedom in prescribing
 122 the two families of planes $U(v), V(u)$. However, since we want to achieve a regular parameterization, we have to make sure that
 123 the intersection curves $S \cap U(v)$ form a fibration of the surface, and the same shall be true for the curves $S \cap V(u)$. Moreover,
 124 the two curve sets should intersect transversally. This may not be possible globally.

125 When prescribing the plane families, we have to stay away from the envelope of planes, which is a developable surface that
 126 may degenerate to a single straight line (see Fig. 4). Recall that the envelope surface E_U of a family of planes

$$U(v) : u_0(v) + u_1(v)x + u_2(v)y + u_3(v)z = 0,$$

127 where $u_i(v)$ are C^1 -functions of v , is computed by intersecting the planes with the derivative planes (dots indicating derivatives
 128 with respect to v),

$$\dot{U}(v) : \dot{u}_0(v) + \dot{u}_1(v)x + \dot{u}_2(v)y + \dot{u}_3(v)z = 0.$$

129 The intersection lines $U(v) \cap \dot{U}(v)$ form a developable ruled surface E_U ; for details we refer to [29].

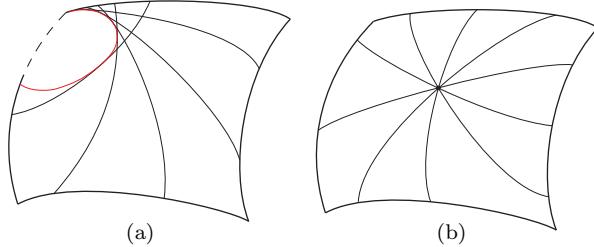


Figure 4: To construct a regular parameterization of a surface with a family of planar parameter lines, one has to avoid the envelope of planes (a). A case which may still be useful despite its singularity is the one where all intersection curves pass through a point (b).

130 Totally analogous statements hold in the discrete setting. If we would like to approximate a surface S with a P-mesh or
 131 PP-mesh M , we have to make sure that the intersection lines of consecutive planes in a family, e.g. $U_j \cap U_{j+1}$ do not intersect
 132 the reference surface S .

133 2.1.2. Conjugate parameterizations

134 A PQ mesh M approximating a surface S is closely tied to the curvature behavior of S . Assuming that the parameter
 135 lines of M are fair and thus represent a proper discretization of a smooth surface parameterization, a PQ mesh M is a *discrete*
 136 *conjugate parameterization* [5]. Hence, it is a discrete counterpart to a conjugate parameterization $\mathbf{x}(u, v)$ of a smooth surface
 137 S , that is characterized by linear dependence of the first partial derivatives $\mathbf{x}_u, \mathbf{x}_v$ and the second mixed derivative \mathbf{x}_{uv} . There
 138 are infinitely many conjugate parameterizations of a surface S . They are also called *conjugate curve networks*.

139 To get such a conjugate network one can prescribe one family of curves and get the other family as follows: At each point
 140 of S , one computes the conjugate direction to the tangent of the given curve family. This computation can be performed with
 141 help of the second fundamental form of $\mathbf{x}(u, v)$ [10]: If $\mathbf{n}(u, v)$ denotes a unit normal vector field of $\mathbf{x}(u, v)$, one forms the inner
 142 products with the second derivatives of \mathbf{x} ,

$$L = \mathbf{x}_{uu} \cdot \mathbf{n}, M = \mathbf{x}_{uv} \cdot \mathbf{n}, N = \mathbf{x}_{vv} \cdot \mathbf{n}.$$

143 Then, two tangent vectors $\mathbf{t}_i = a_i \mathbf{x}_u + b_i \mathbf{x}_v$, $i = 1, 2$ are conjugate if

$$La_1a_2 + M(a_1b_2 + a_2b_1) + Nb_1b_2 = 0. \quad (1)$$

144 Hence, if one direction is given, we can compute the conjugate direction. Integration of the field of conjugate directions
 145 to the tangents of the first curve family yields the second curve family and thus a conjugate curve network. However, there
 146 is a caveat: at hyperbolic points, there are the so-called asymptotic directions (with vanishing normal curvature). They are
 147 self-conjugate and therefore have to be avoided if one wants to compute a regular curve network. In fact, for applications one
 148 has to avoid small intersection angles between the two curve families as well and thus stay sufficiently far away from asymptotic
 149 directions.

150 2.2. Approximation with P+PQ meshes

151 From what has been said above, it is clear that *one can (at least locally) approximate any surface S by a PQ mesh which*
 152 *has one family of planar polylines (P+PQ mesh)*. One has to use a guiding frame field of conjugate parameterization on S in
 153 which one tangent vector lies in a family of planes, say $U(v)$. The difficulty arises in areas of negative Gaussian curvature,
 154 where one has to choose these planes so that they are nowhere tangent to an asymptotic direction.

155 2.2.1. Rationalization workflow

156 We deal with the approximation of a given reference surface S by a P+PQ mesh with regular combinatorics. In architecture,
 157 such a process is called rationalization of a design to simplify its fabrication. For a given surface (represented as a dense triangle
 158 mesh), we assign a conjugate direction pair to each face barycenter. One direction t_1 lies in a plane of the family $U(v)$ still to
 159 be defined, the other direction t_2 is conjugate to it. To compute the conjugate direction, we use the *jet fit* method of Cazals
 160 and Pouget [7] to numerically calculate the first and second derivatives. The rationalization workflow is described as below and
 161 illustrated in Figure 5.

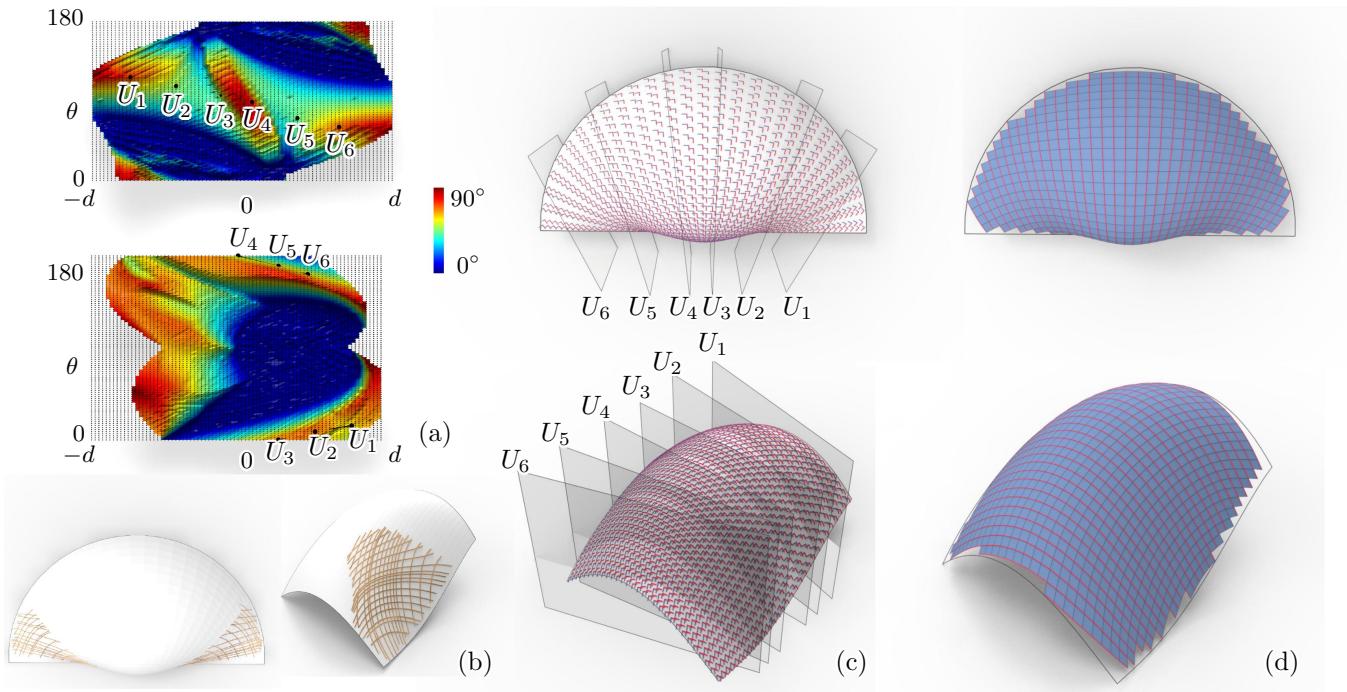


Figure 5: Two examples for approximation with P+PQ meshes (a) The user can select points of vertical planes on the color coded quality function, noting the periodicity with respect to the angle θ . (b) The curves on the surfaces show the asymptotic nets as additional support. Chosen planes should not be tangent to asymptotic curves. (c) Interpolation of the selected planes and computation of conjugate directions to those in the planes yield a conjugate direction field. (d) Based on this conjugate field, we approximate the surface by remeshing and optimization. The red polylines are the planar ones.

162 **Selection of planes.** These planes $U(v)$ intersecting the surface with one family of planar parameter lines should satisfy two
 163 conditions:

- 164 • The envelope of the planes should not intersect the surface, as discussed in 2.1.1.
- 165 • The intersecting curve should avoid asymptotic directions.

166 For initialization, we start with a special case where the planes containing the parameter lines are vertical. This is an
 167 assumption for structural reasons and can later be changed in the post-optimization procedure. In a top view, the planes
 168 appear as straight lines. Hence, a proper choice of a family of planes requires the proper choice of a family of straight lines in
 169 the top view.

170 We provide a color coded quality function as a visual tool to guide the design of planes. As stated above, planes $U(v)$ are
 171 assumed to be vertical (parallel to the z -axis) and of course have to intersect S . We sample this 2-parameter set of planes by the
 172 signed distance d to the origin (barycenter of S) and direction angle $\theta \in [0, \pi]$ against the x -axis. Hence, each possible plane U
 173 is represented by a point on the (d, θ) plane. For each triangle $\Delta_i \subset S$ that intersects U , we take t_1 parallel to the intersection
 174 line $\Delta_i \cap U$, compute the conjugate direction t_2 and angle α_i between t_1, t_2 . Then the minimal angle $f(U) = \min(\alpha_i)$ serves
 175 as a quality measure for U to color the corresponding point (d, θ) .

The color coded quality function over the (d, θ) -plane guides the user to select a few good positions for planes (Fig. 5). In addition, we provide the user with the net of asymptotic curves (Fig. 5). Planes have to be selected so that they are nowhere nearly tangent to an asymptotic curve. The user-defined planes are then interpolated to obtain a denser set of planes.

Remeshing and optimization. Based on the selection of planes, a frame field of pairs of conjugate directions are computed. We re-mesh S by this guiding frame field using the implementation of mixed integer quadrangulation (MIQ) [6] in LIBIGL [12]. Using the remeshed structure as an initialization, we employ an optimization procedure to refine the shape by keeping the essential constraints, such as P, PQ and proximity to the original shape (see Section 5).

2.2.2. Discussion

Fig. 5 shows two examples in which the sampling method helps the user to find initial planes. However, this boundary-to-boundary plane arrangement can fail when the surface has a complicated behavior of asymptotic directions. For such a more difficult situation, we can approximate the given shape by PQ-meshes with one family of piecewise planar polylines. A solution may incorporate the choice of a combinatorial singularity, for which we do not have an algorithmic solution so far. An example for that is shown by Fig. 6.

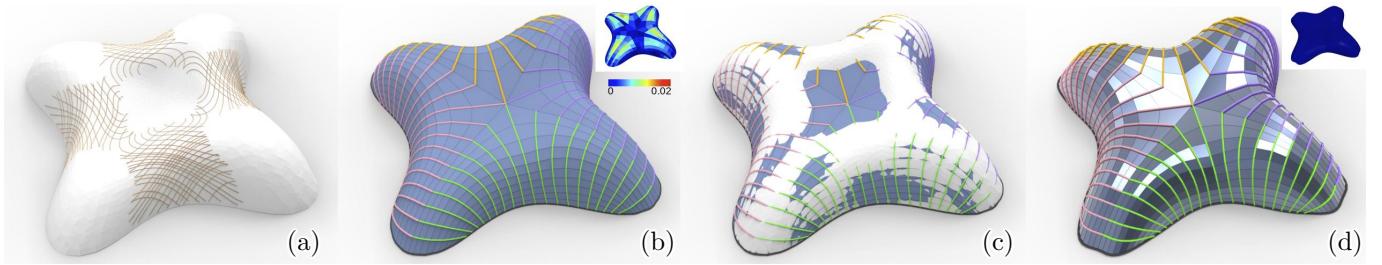


Figure 6: P+PQ mesh approximation for the top of the Lilium Tower. (a) The complicated asymptotic curves lead to no regular settings of planes with good quality. (b) However, we can approximate it by pieces of P+PQ meshes assembled around a central combinatorial singularity of valence 8. One half of the polylines are colored, indicating their planarity in piecewise. In addition, the color-bar shows that the error of face planarity is within the range of 2%, which is measured by the distance between diagonals divided by the average length of diagonals in each face. (c) The optimized P+PQ mesh approximates the input shape well. (d) We can further optimize the faces to be accurately planar at the expense of a larger deviation from the target shape.

3. Combining planar polylines and funicularity

3.1. Static equilibrium without vertical loads

In this subsection, we introduce some interesting connections of the present topic to geometry, which arise if one asks for static equilibrium. This is a natural question for the architectural application. However, initially we deal here only with the simplified case of no external loads, even neglecting the weight of the structural elements.

At first, we consider a supported boundary. Now, each inner edge, for simplicity called $\mathbf{v}_i\mathbf{v}_j$, gets a force density w_{ij} such that $w_{ij}(\mathbf{v}_i - \mathbf{v}_j)$ is the force exerted at \mathbf{v}_i and the opposite force $w_{ij}(\mathbf{v}_j - \mathbf{v}_i)$ is exerted at \mathbf{v}_j . Tensile forces (like in cables) have $w_{ij} < 0$, while compressive forces belong to $w_{ij} > 0$. Equilibrium is characterized by a vanishing resulting force at each inner node \mathbf{v}_i ,

$$\sum w_{ij}(\mathbf{v}_i - \mathbf{v}_j) = 0, \quad (2)$$

where the sum is over all j characterizing the four connected neighbors of \mathbf{v}_i .

For an unsupported boundary, one also has forces in the boundary edges and this condition is applied at the non-supported boundary vertices. This shows the well-known fact that *an unsupported boundary polyline is a discrete asymptotic curve of the surface*: The three edges meeting at an unsupported boundary vertex \mathbf{v}_i must be coplanar and the two boundary edges span the discrete osculating plane of the boundary. Now the claim follows from the fact that the osculating planes of an asymptotic curve on a surface are tangent to that surface. The corner vertices, where only two (non-collinear) edges meet, have to be supported.

It is a remarkable result that such equilibrium structures are even *invariant under projective transformations* [31]. However, forces and moments have to be transformed like line coordinates. Of course, projective maps also keep the planarity of faces and parameter lines.

According to the principles of graphic statics, one can now form a so-called *reciprocal force diagram*. This is the mesh of forces. Since the forces at a vertex of M sum up to zero, they can be arranged as the edge vectors of a closed quad, and all these quads form the reciprocal mesh M^* . It is known that M^* is also in equilibrium with M as its reciprocal force diagram [16, 31].

Corresponding edges in a reciprocal mesh pair (M, M^*) are parallel. However, edges through a vertex of M belong to edges in a face of M^* and vice versa. One speaks of *reciprocal parallel meshes*. We see that the reciprocal parallel mesh M^* of a PQ mesh M is not a PQ mesh, but a mesh with planar vertex stars, a so-called A-net. A-nets are discrete asymptotic parameterizations and have been studied extensively in discrete differential geometry [5].

Planarity of parameter lines is not preserved when switching from M to M^* . A planar polyline in M (say, in a plane $U(v)$) corresponds to a strip of edges in M^* which are parallel to the plane $U(v)$ and connect the vertices of two neighboring parameter lines of M^* (see Fig. 7). These connecting edges can be viewed as rulings of a so-called conoidal ruled surface.

Hence, the reciprocal mesh M^* to a PP+PQ mesh M in equilibrium (without external forces) is a *discrete affine minimal surface* (Fig. 7). It is a discrete version of smooth surfaces which possesses an asymptotic parameterization with the following property: Along each asymptotic curve, the other asymptotic tangents form a conoidal ruled surface (i.e., they are parallel to a plane). According to W. Blaschke [4], this characterizes negatively curved affine minimal surfaces.

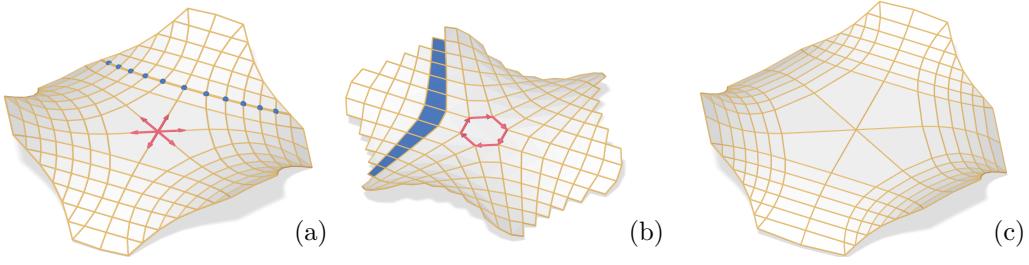


Figure 7: Quad meshes in static equilibrium. (a) A PP+PQ mesh M in static equilibrium. (b) Its reciprocal force diagram M^* is a discrete affine minimal surface. Red: Forces through a vertex of M correspond to edges in a face of M^* . Blue: A planar polyline in M corresponds to a strip of edges in M^* . These edges are parallel to the plane containing the corresponding polyline in M . (c) A parallel mesh \bar{M} of M is also a PQ+PP mesh and in equilibrium with M^* as reciprocal force diagram.

There is another, simpler type of mesh parallelism which plays a role in our context as well. A pair of combinatorially equivalent meshes M, \bar{M} is called parallel (or related by a Combescure transformation [5]) if corresponding edges are parallel. This is only possible for PQ meshes. Since planar parameter lines are also kept under this mesh parallelism, we can state that PQ meshes with planar parameter lines are invariant under mesh parallelism. Obviously, also static equilibrium is preserved under parallelism, since we can take the forces from the original mesh M , i.e., keep the reciprocal force diagram M^* unchanged.

3.2. Static equilibrium with vertical loads

In this subsection, we turn to a more practical problem and discuss the rationalization of a given shape by a funicular P-mesh (P+F mesh) considering vertical loads. It turns out that this approximation task is closely related to the one addressed in 2.2, if we consider the so-called Airy stress surface, explained below.

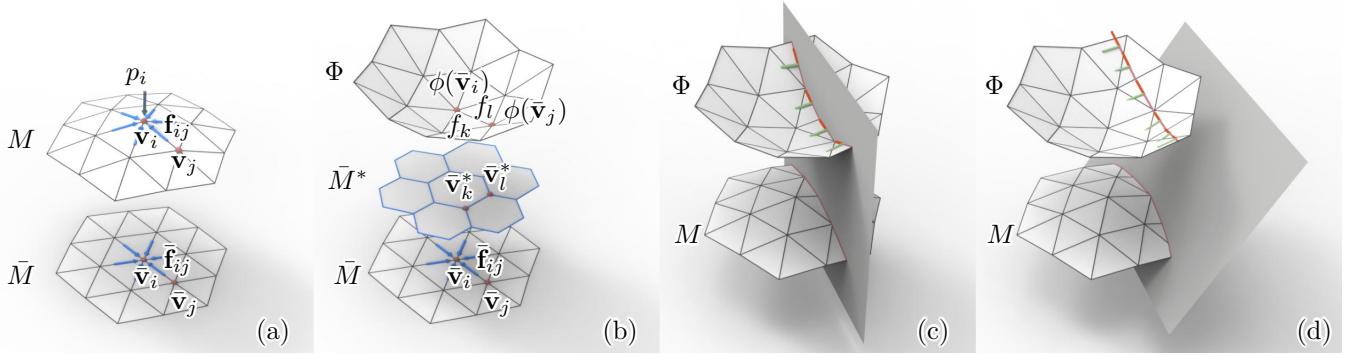


Figure 8: Airy stress surface and using it for remeshing with a P+F mesh. (a) A mesh M in static equilibrium under vertical load with supported boundaries, and its projection \bar{M} onto the xy plane. (b) The horizontal equilibrium of \bar{M} forms a force diagram \bar{M}^* . It is a dual mesh whose edges are constructed from the forces of \bar{M} and rotated by 90 degrees [16]. The Airy polyhedron Φ has the same top view \bar{M} as M . It has planar faces whose gradients are defined by the coordinates of the corresponding vertex of \bar{M}^* . (c) In the special case where M is rationalized towards a P+F mesh with vertical planar polylines, we perform P+PQ remeshing of Φ and map back to M . (d) When the planar polylines on M are not confined to be vertical, we project them in vertical direction to Φ , compute conjugate directions there, and use the resulting cross field on Φ for remeshing.

As shown in Figure 8, for a mesh M in static equilibrium under vertical loads with supported boundary, each unsupported vertex \mathbf{v}_i satisfies an equilibrium equation (assuming a vertical z -axis):

$$\sum w_{ij}(\mathbf{v}_i - \mathbf{v}_j) = \begin{bmatrix} 0 \\ 0 \\ p_i \end{bmatrix}, \quad (3)$$

where the sum is over all j characterizing the connected neighbors of \mathbf{v}_i and p_i is the vertical load on vertex \mathbf{v}_i .

Considering just the x and y coordinates in (3) shows that the top view \bar{M} of M in the xy plane is in equilibrium without external forces and with forces $\bar{\mathbf{f}}_{ij} = w_{ij}(\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j)$ in the edges. Here $\bar{\mathbf{v}} = (v_x, v_y)$ denotes the top view of a point $\mathbf{v} = (v_x, v_y, v_z)$. The forces $\bar{\mathbf{f}}_{ij}$ at each vertex $\bar{\mathbf{v}}_i$ form a closed planar polygon of the reciprocal force diagram \bar{M}^* . Here we apply a 90 degree rotation to the force vectors $\bar{\mathbf{f}}_{ij}$ in order to obtain the edge vectors of \bar{M}^* .

Based on \bar{M}^* , one can construct a polyhedral stress mesh (so-called Airy stress surface) $\Phi = (x, y, \phi(x, y))$, which shares the same projection \bar{M} in the xy plane. For a planar face f_k of Φ , its gradient is $\nabla\phi|_{f_k} = (x_k^*, y_k^*)$, where (x_k^*, y_k^*) are the coordinates of vertex $\bar{\mathbf{v}}_k^*$ in the force diagram. In other words, $\bar{\mathbf{v}}_k^* = (n_k^x/n_k^z, n_k^y/n_k^z)$, where $\mathbf{n}_k = (n_k^x, n_k^y, n_k^z)$ is a normal of f_k .

Each oriented edge $\vec{e}_{ij} = (\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j)/||\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j||$ on the xy plane has two adjacent faces f_k and f_l at its corresponding edge on the Airy mesh Φ . The force along this edge \vec{e}_{ij} is computed by

$$\beta^{is}(\vec{e}_{ij}) = J(\nabla\phi|_{f_k} - \nabla\phi|_{f_l}) \cdot \vec{e}_{ij}, \quad (4)$$

where $J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ describes the counterclockwise 90° rotation in the xy plane. This force is equal to the signed isotropic angle between the planes of faces f_k and f_l . An introduction to isotropic geometry is found in [27] and for detailed discussions of the geometry of equilibrium, we refer to [24, 22, 21, 43].

As indicated in Figure 8, we may be given a funicular design surface, represented by a triangle mesh M that is in equilibrium with the given loads and thus possesses an associated stress surface Φ . We want to represent the design surface by a funicular quad mesh M_1 , which in addition has a family of planar polylines. Letting aside the latter property for a moment, we mention the main property of the Airy surface Φ which helps to solve our problem: *A funicular quad mesh M_1 representing the design surface corresponds (by vertical projection) to a PQ mesh representing Φ .* PQ-remeshing of Φ is easy if Φ is free of asymptotic directions, which happens if all forces are compressive or all are tensile. The harder case is the one where both compression and tension occur, since this corresponds to negatively curved regions of the stress surface.

We see that rationalizing a given shape M by a P+F mesh is very closely related to the already discussed approximation with a P+PQ mesh (subsection 2.2). A P+F mesh with vertical planar polylines should have a corresponding Airy polyhedron which is a P+PQ mesh with vertical planar polylines. Hence, this rationalization task can be started with a P+PQ remeshing approach of the corresponding Airy surface Φ .

For more general designs, the planar polylines of the P+F mesh need not be vertical. In this case, we have to select planes for the parameter lines on M so that their intersections with the design surface correspond to curves on Φ which avoid asymptotic directions. We then project the planar sections on M vertically to Φ and compute the conjugate field on Φ . After remeshing the Airy surface Φ according to this conjugate field, it is already close to a PQ mesh. We can further optimize the face planarity with proximity to the original Airy surface Φ . Then we vertically map the polylines back onto the target M , with the isotropic angle between faces of the Airy mesh as the initial force value. With this initialized mesh and forces, we can find a P+F mesh in close proximity to the given shape by optimization (for details of optimization, see Section 5).

For some special cases where the designed parameter lines are close to the conjugate nets on both Airy surface and target surface, it is possible to rationalize the surface as a P+PQ+F mesh. A self-Airy surface M , which agrees (up to scaling in z -direction) with its Airy surface Φ , is obviously one category of surfaces that can be approximated in this way (Figure 9). For the construction of self-Airy surfaces, we refer to [22, 21]. Another example of P+PQ+F rationalization is shown in Figure 10. More generally, Figure 11 shows an example of rationalization by a P+F mesh where the Airy surface is not convex, so that remeshing should avoid touching the asymptotic curves.

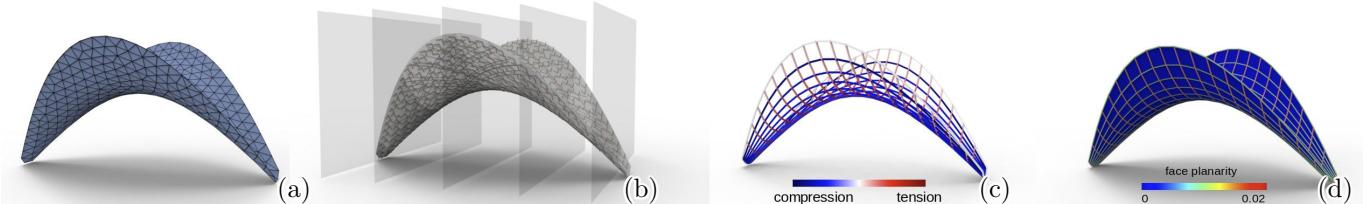


Figure 9: Rationalization of a self-Airy surface by a P+PQ+F mesh. (a) A given self-Airy surface, as shown in [8], is represented by a triangle mesh. (b) Remesh the Airy mesh towards a P+PQ mesh based on the conjugate field guided by the cutting planes. (c) and (d) show the force distribution, planar polylines (actually both families are planar, so we have a PP+PQ+F mesh) and planar quad panels after optimization.

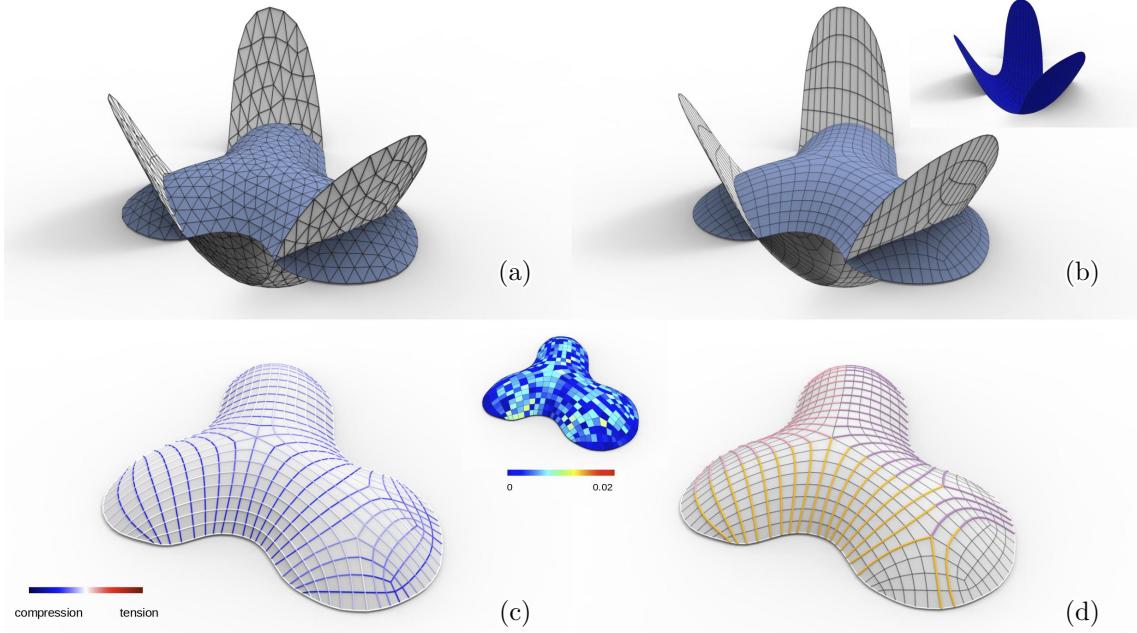


Figure 10: Rationalization with a P+PQ+F mesh. (a) A given surface in static equilibrium (in blue) and its Airy stress surface (in white) are represented by triangle meshes. (b) Design the planes containing planar polylines on the target surface and remesh the Airy mesh along a conjugate field. (c) and (d) show the force distribution, piecewise planar polylines (in colors) and near planar quad panels after optimization.

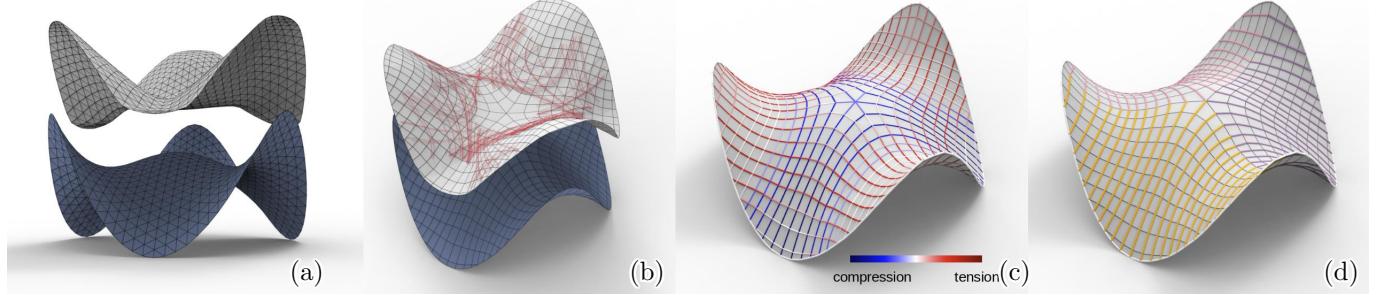


Figure 11: Rationalization with a P+F mesh. (a) A given surface in static equilibrium (in blue) and its Airy stress surface (in white) are represented by triangle meshes. (b) Design the planes containing planar polylines on the target surface and remesh the Airy mesh by a conjugate field without touching the asymptotic curves (in red). (c) and (d) show the force distribution and piecewise planar polylines (in colors) after optimization.

269 4. Meshes with orthogonal planar parameter lines

270 4.1. Smooth surfaces

271 Let us first provide a view into classical geometric research on surfaces which carry two orthogonal families of planar curves.

272 T. Ivey [11] showed that the only surfaces which carry two orthogonal families of circles are Dupin cyclides. The circles are
273 either principal curvature lines or form an angle of 45 degrees with those (*Villarceau circles*). In the latter case, the surfaces
274 are related to a Willmore torus (torus of revolution with radii 1 and $\sqrt{2}$) by a Möbius transformation. The Villarceau circles
275 on other ring cyclides do not intersect under a right angle, but under a constant angle.

276 Another special case of surfaces with orthogonal planar parameter lines are those where one family of curves are straight.
277 These ruled surfaces have been completely classified by H. Sachs [30]. If they are developable ruled surfaces and possess a
278 family of curves orthogonal to the rulings, they are either cylinders or developable surfaces of constant slope. In both cases
279 the planes $U(v)$ of the curves orthogonal to the rulings are parallel and thus these are also the surfaces with straight curves of
280 steepest descent. The surfaces intersect the planes $U(v)$ under a constant angle. For a detailed study of these surfaces, we also
281 refer to [29].

282 Generalizations of developable surfaces of constant slope are surfaces with an orthogonal net of planar curves, where one
283 family of parameter lines lie in parallel planes. We may imagine these planes as horizontal. In an architectural application
284 they could form the floor slabs of a building. Then the orthogonal curves are curves of steepest descent. There has been a
285 number of contributions on surfaces with planar curves of steepest descent, especially by W. Wunderlich [45, 46, 48, 47] and
286 more recently by H. Trautwein [42]. If, in addition the second family of curves lies in vertical planes, we arrive at the familiar

287 moulding surfaces, which are special cases of surfaces with planar curvature lines which have recently been discussed in detail
 288 in connection with the architectural application [20, 39, 13].

289 **Surfaces with parabolas as curves of steepest descent.** Since our computational approach is based on numerical
 290 optimization, it is important to have good starting meshes for further manipulation. In one of the above mentioned papers
 291 on surfaces with planar curves of steepest descent we find a sufficiently large set of examples to get started with. These are
 292 the surfaces which possess parabolas as curves of steepest descent. W. Wunderlich [46] provided an elegant construction of all
 293 these surfaces through an appropriate transformation of developable surfaces of constant slope (for the latter, see the detailed
 294 treatment in [29]). It is sufficient here to assume slope 1. These surfaces Φ^* are envelopes of a one-parameter family of planes
 295 which are inclined against the plane $z = 0$ under the angle $\pi/4$:

$$T(v) : x^* \cos v + y^* \sin v - z^* = h(v).$$

296 Here, the arbitrary function $h(v)$ is the support function of the intersection of Φ^* with the plane $z = 0$. To compute the
 297 envelope of tangent planes $T(v)$, we have to compute the derivative planes $\dot{T}(v) : -x^* \sin v + y^* \cos v = \dot{h}(v)$ and intersect with
 298 $T(v)$. This yields the developable surface Φ^* of constant slope:

$$\mathbf{x}^*(u, v) = ((h + u) \cos v - \dot{h} \sin v, (h + u) \sin v + \dot{h} \cos v, u).$$

299 It is convenient to consider complex numbers $\zeta^* = x^* + iy^*$, since then the first two coordinate functions are simply

$$\zeta^* = (h + i\dot{h} + u)e^{iv}.$$

300 Note that the surface Φ^* has straight u -lines which intersect the v -lines in planes $z = u$ under a right angle. The same right
 301 angle appears in the top view (projection into the plane $z = 0$). Following Wunderlich [46], we now apply the conformal map
 302 $\gamma : \zeta^* \mapsto \zeta = (\zeta^*)^2$ to the first two coordinates and keep the third coordinates z^* unchanged. This yields the surface Φ :

$$\Phi : \zeta = x + iy = (h + i\dot{h} + u)^2 e^{2iv}, z = u. \quad (5)$$

303 The conformal map γ transforms straight lines to parabolas with focal point at the origin. The simple 3D extension maps
 304 general straight lines in space to parabolas whose axes are parallel to the plane $z = 0$. The v -lines in planes $z = u$ are also
 305 transformed by γ and the right angle to the u -lines is preserved. Hence we have a surface Φ all whose curves of steepest descent
 306 (u -lines) are parabolas with horizontal axis. Wunderlich shows that the other essentially different case is the one with $z = u^2$
 307 instead of $z = u$. Now all parabolas are tangent to the plane $z = 0$ along a curve and their axes possess constant slope 1. We
 308 can set $z = a_0 u^2 + a_1 u$ and also obtain surfaces with parabolas as curves of steepest descent. This amounts to applying the
 309 transformation

$$(x^*, y^*, z^*) \mapsto ((x^*)^2 - (y^*)^2, 2x^*y^*, a_0(z^*)^2 + a_1z^*) \quad (6)$$

310 to Φ^* .

311 In order to design Φ , it is better to prescribe its curve c in the plane $z = 0$ (see Figures 12). Then, one applies the inverse
 312 conformal map $\zeta \mapsto \sqrt{\zeta}$ to it, obtaining a curve c^* on a developable surface Φ^* of constant slope 1. We need not use the support
 313 function to compute Φ^* , but just consider the fact that the rulings of Φ^* have the normals of c^* as their top view. Hence, if c^*
 314 is given in a parameterization $c^*(v) = (c_1^*(v), c_2^*(v))$, the surface Φ^* can be written as

$$\mathbf{x}^*(u, v) = (c_1^* - \frac{u}{\sqrt{(\dot{c}_1^*)^2 + (\dot{c}_2^*)^2}} \dot{c}_2^*, c_2^* + \frac{u}{\sqrt{(\dot{c}_1^*)^2 + (\dot{c}_2^*)^2}} \dot{c}_1^*, u).$$

315 Transforming back via a map of the form (6) yields the surface Φ . In the following, surfaces with parabolas as curves of steepest
 316 descent will be called *Wunderlich surfaces*.

317 **Partial differential equation characterizing surfaces with orthogonal planar parameter lines.** Preparing for the
 318 construction of orthogonal PP meshes, we first describe here the continuous form. For that, we prescribe two families of planes
 319 $U(v), V(u)$ and consider the intersection lines $l(u, v) = U(v) \cap V(u)$. This 2-parameter family of lines (line congruence; see [29])
 320 may be written with a guiding surface $\mathbf{s}(u, v)$ and direction vectors $\mathbf{d}(u, v)$ as

$$l(u, v) : \mathbf{x} = \mathbf{s}(u, v) + w\mathbf{d}(u, v).$$

321 We have to find $w = w(u, v)$ so that the parameter lines on the surface $\mathbf{x}(u, v) = \mathbf{s}(u, v) + w(u, v)\mathbf{d}(u, v)$ are orthogonal. The
 322 orthogonality condition $\mathbf{x}_u \cdot \mathbf{x}_v = 0$ leads to a nonlinear first order PDE for the function $w(u, v)$,

$$(\mathbf{s}_u + w_u \mathbf{d} + w \mathbf{d}_u) \cdot (\mathbf{s}_v + w_v \mathbf{d} + w \mathbf{d}_v) = 0. \quad (7)$$

323 It is possible to find explicit solutions for special cases such as the one where both plane families are pencils of planes. However,
 324 this would lead too far away from the main purpose of this paper and thus we defer those results to another publication. Here,

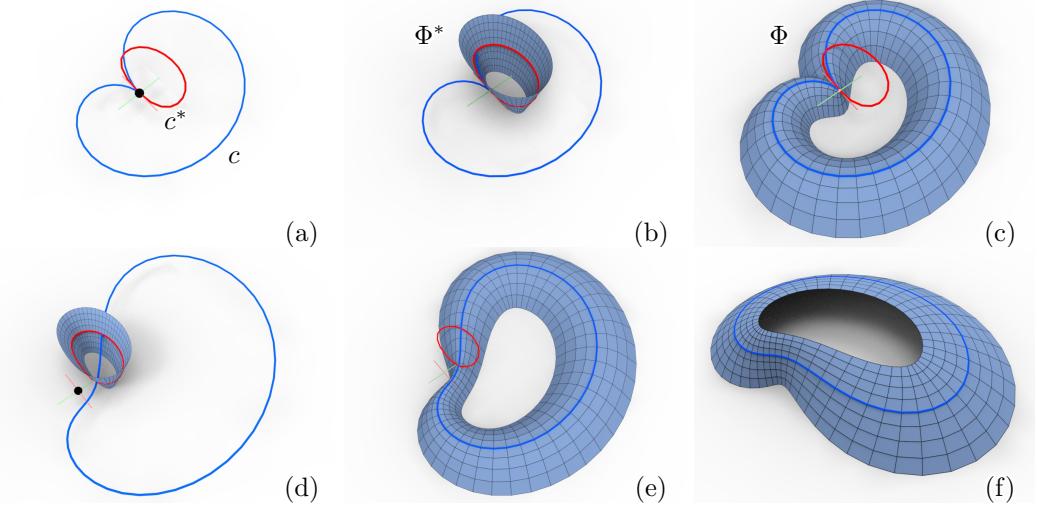


Figure 12: Construction of Wunderlich surfaces. (a) One chooses a curve c (blue) in the plane $z = 0$ and applies the conformal map $\zeta \mapsto \sqrt{\zeta}$ to it. This yields the curve c^* (red). (b) A developable surface Φ^* of constant slope is computed through c^* . (c) Application of a map (6) (here $(a_0, a_1) = (1, 0)$) yields a Wunderlich surface Φ through c . (d,e) The result depends on the location of the origin in the base plane. Here, we fixed c^* and changed the origin. (f) A Wunderlich surface constructed on the same base curve c , but with $(a_0, a_1) = (-0.3, 1)$.

325 we focus on the discrete case and note that the construction provided below is an algorithm to find numerical solutions of this
 326 PDE.

327 4.2. Discrete surfaces

328 We use special quad meshes as discrete models of orthogonal surface parameterizations. As explained in Fig. 13, we require
 329 each face in the mesh to have diagonals of equal length. This discrete orthogonality constraint is now used for a numerical
 330 solution of equation (7) and later in the global optimization of Section 5.

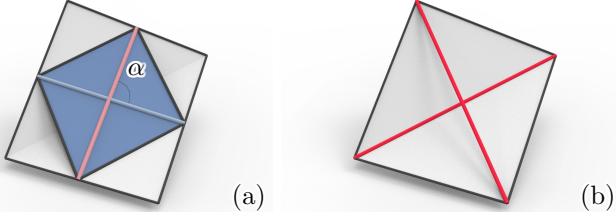


Figure 13: Discrete orthogonal surface parameterizations. (a) Orthogonality is defined per face. In each face, we require orthogonal connecting lines of opposite edge midpoints ($\alpha = 90^\circ$). Edge midpoints are vertices of a parallelogram (blue). Since it has orthogonal diagonals, it is a rhombus. (b) The edges of that rhombus are parallel to the diagonals (red) of the quad and have half of the diagonals length. Hence, discrete orthogonality is formulated via diagonals of equal length in each face of the quad mesh.

331 4.2.1. Form-finding using propagation for initialization

332 We are now ready to present a numerical solution of PDE (7) to obtain discrete versions of surfaces with orthogonal planar
 333 parameter lines ($PP+90^\circ$ meshes). It is one way to come up with initial shapes that can later be edited by global optimization.

334 Firstly, we prescribe the planes of parameter lines. One family of parameter lines lie in planes U_1, U_2, U_3, \dots and the other
 335 family in planes V_1, V_2, V_3, \dots . This constrains the vertices $\mathbf{v}_{i,j}$ of the mesh to the intersection lines $l_{i,j} = U_i \cap V_j$. In addition
 336 to the planarity of parameter lines, we aim at their orthogonality. This requires equal-length diagonals in the quads of the
 337 mesh, as shown in Fig. 13.

338 We can construct a $PP+90^\circ$ mesh to the provided data from one boundary polyline by a *propagation algorithm* as follows:
 339 We prescribe a boundary polyline on plane U_1 and an initial guess of the next vertex (green) of the other boundary on plane
 340 V_1 as shown in Fig. 14. We can then find the positions of the vertices for the next polyline on plane U_2 via orthogonality.
 341 This propagation is achieved by progressively intersecting a sphere (radius equals a known diagonal length) with a line $l_{i,j}$
 342 (Fig. 14 (a)). The number of intersection points can be 2, 1 or 0. If there are two intersection points, we select the one with
 343 lower fairness energy (fairness is expressed via 2nd differences; see Section 5). If there is no intersection, it is natural to use the
 344 point of $l_{i,j}$ that is closest to the sphere, and thus we get $\mathbf{v}_{i,j}$ by orthogonal projection of the sphere center $\mathbf{v}_{i-1,j-1}$ onto $l_{i,j}$.
 345 Immediately after getting this first version of the mesh polyline in U_2 , we optimize it for fairness keeping the constraints; this

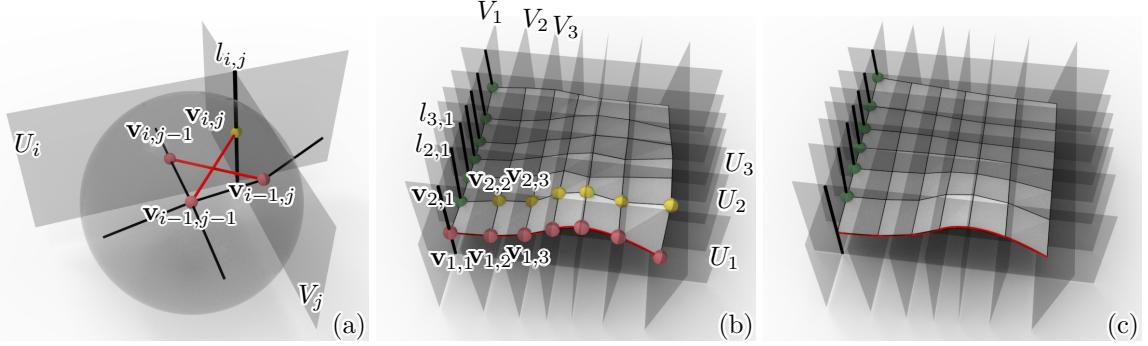


Figure 14: Construction of a PP+90° mesh by propagation. (a) The basic step computes from three known vertices $\mathbf{v}_{i-1,j-1}$, $\mathbf{v}_{i-1,j}$, $\mathbf{v}_{i,j-1}$ to a vertex $\mathbf{v}_{i,j}$ on a line $l_{i,j}$ so that the diagonals in the resulting quad have equal length; this amounts to intersecting a sphere centered at $\mathbf{v}_{i-1,j-1}$ with $l_{i,j}$. (b) Propagation from a prescribed boundary polyline in U_1 to the next polyline in U_2 . (c) Final result after global optimization for fairness.

will in general change the originally chosen point in V_1 . We then repeat the steps described above to find the next polyline on U_3 , and so on. Finally, we employ a global optimization with various considerations from a practical perspective. This depends on the requirements of the application and may allow deviations from the originally fixed plane positions, from precise discrete orthogonality, and from the boundary in U_1 .

Examples of meshes constructed with the propagation algorithm and later modified through global optimization are shown in Fig. 15. In particular, the figure shows that relaxing the orthogonality constraint enlarges the design space so that further constraints, e.g. at boundaries, can be applied.

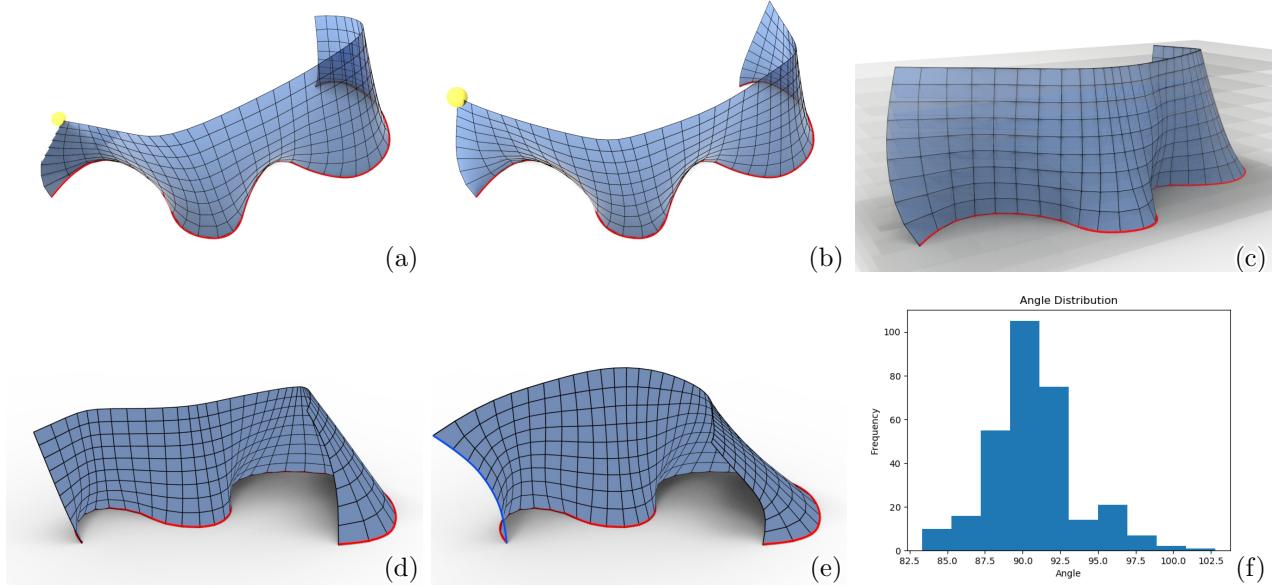


Figure 15: PP+90° meshes found by propagation plus optimization. (a) shows a result of propagation with one prescribed boundary (red) on the ground. (b) The undesirable shape of the surface near the yellow vertex can be improved by changing its position through global optimization keeping the planarity and orthogonality of polylines, the given boundary and fairness. (c) An orthogonal PP-mesh with one family of polylines in horizontal planes, constructed through the same boundary on the ground. (d) Another PP+90° mesh found by propagation plus optimization with a fixed ground boundary. (e) A user may request to control both boundaries (red and blue) for architectural applications. Since a propagation with two fixed boundaries may lose fairness, a soft constraint for orthogonality is applied to find a near orthogonal PP-mesh. (f) shows its angle distribution where the angles are measured between connecting lines of opposite edge midpoints in each face, referring to Fig. 13.

4.2.2. Form-finding using Wunderlich surfaces for initialization

A PP+90° mesh with one family of polylines in horizontal planes can be obtained by sampling a Wunderlich surface. Its construction can start from a based curve c , as illustrated in Fig. 12. Sampled Wunderlich surfaces are just a very special subset of PP+90° meshes. They may not fulfil our discrete orthogonality constraint precisely. This does not hurt in the subsequent editing through optimization (see Section 5), which we can apply in order to enlarge the design space. Figs. 16 and 17 show examples.

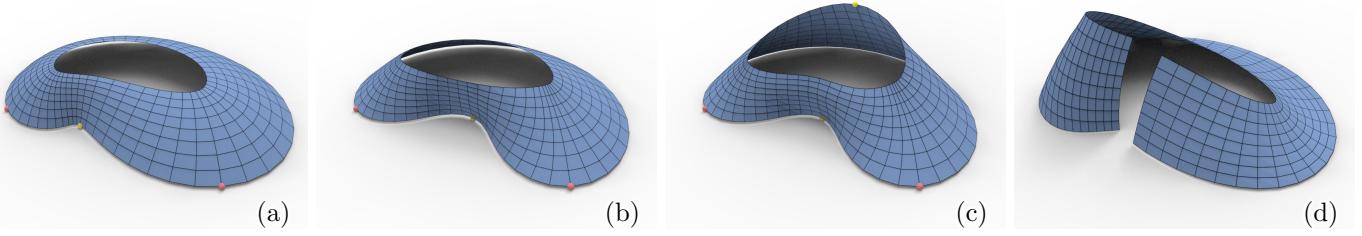


Figure 16: Designing PP+90° meshes, starting from a sampled Wunderlich surface. (a) Original mesh sampled from the Wunderlich surface of Fig. 12 (f). (b) Editing to find a new PP+90° mesh with one family of polylines in horizontal planes. (c) Editing to find a general PP+90° mesh without preserving the horizontally parallel polylines. In these examples, pink vertices are fixed and editing relocates the yellow vertices. (d) A result of several editing operations, after cutting the surface open along a profile.

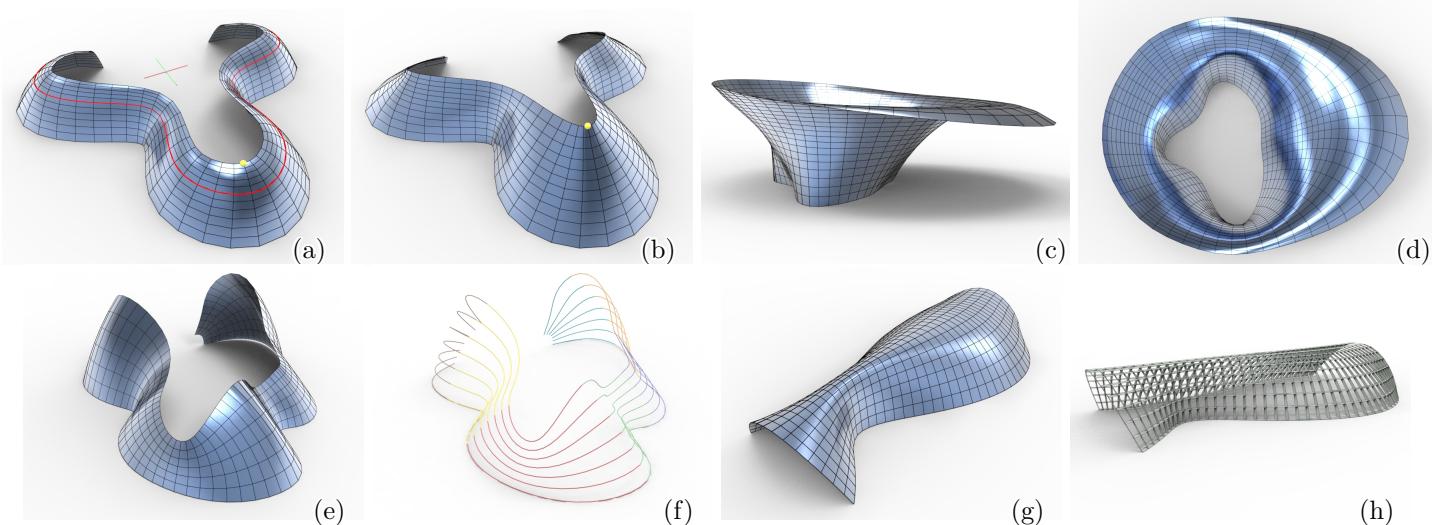


Figure 17: PP+90° meshes designed via editing Wunderlich surfaces. (a) Sampled Wunderlich surface, designed through the red curve. (b) A PP+90° mesh obtained from (a) by fixing the bottom boundary and lifting the yellow vertex. (c) and (d) show designs obtained by editing a Wunderlich surface through a user-defined bottom curve. (e) The design freedom is enlarged by requiring one family of parameter lines to be just piecewise planar; the pieces are colored in (f). (g) shows another design and its support structures from planar long-range beams in (h).

359 4.3. Discussion

360 Both of the proposed methods are suitable for architectural design, as one can easily start with prescribing a base/boundary
 361 curve on the ground. The Wunderlich-based initialization has more control on the curves in the steepest descent direction,
 362 which however are restricted to parabolas. In contrast, the propagation-based initialization is more flexible, since there is
 363 no restriction on the planes carrying the parameter lines. However, propagation is not guaranteed to yield useful results for
 364 certain choices of a boundary polyline and planes of parameter lines. Overall, both approaches are able to provide a reasonable
 365 initialization for the editing with optimization.

366 5. Optimization

367 As already mentioned in Sections 2, 3 and 4, numerical optimization is used to refine meshes and to broaden the design space
 368 with shape editing. For the computation of constrained meshes, we use the numerical optimization algorithm proposed by Tang
 369 et al. [38]. It is an appropriately regularized Gauss-Newton algorithm. An important ingredient for the high performance is the
 370 use of constraints that are at most quadratic. In many cases, this can be achieved by the introduction of auxiliary variables.
 371 Otherwise, one uses geometrically motivated simplifications and may take certain values for variables from the previous iteration.
 372 We do not consider this a main contribution, but just provide the necessary details for a successful implementation.

373 5.1. Constraints

374 *Planar Polyline constraints.* To construct a P-mesh or a PP-mesh, one or both families of parameter lines have to be planar.
 375 Planarity of a polyline p is achieved by using the normal vector \mathbf{n}_p of its plane as an additional variable and requires orthogonality
 376 of all edges of p to \mathbf{n}_p . To avoid that during optimization \mathbf{n}_p tends to zero and thus the orthogonality constraint is fulfilled in a
 377 trivial way, we require a normalized vector \mathbf{n}_p . If p contains l_p vertices, $\mathbf{v}_{p0}, \mathbf{v}_{p1}, \mathbf{v}_{p2} \dots \mathbf{v}_{p,l_p-1}$, this yields the planarity energy

$$E_{PP} = \sum_{p=0}^{|P|-1} \sum_{j=0}^{l_p-2} (\mathbf{n}_p \cdot (\mathbf{v}_{pj} - \mathbf{v}_{p,j+1}))^2 + \sum_{p=0}^{|P|-1} (\mathbf{n}_p \cdot \mathbf{n}_p - 1)^2, \quad (8)$$

378 where $|P|$ represents the number of polylines in the mesh that need to be planar.

379 *PQ constraints.* Analogously, we express planarity of a face i with vertices $\mathbf{v}_{i0}, \mathbf{v}_{i1}, \mathbf{v}_{i2}$ and \mathbf{v}_{i3} as

$$E_{PQ} = \sum_{i=0}^{|F|-1} \sum_{j=0}^3 (\mathbf{n}_i \cdot (\mathbf{v}_{ij} - \mathbf{v}_{ik}))^2 + \sum_{i=0}^{|F|-1} (\mathbf{n}_i \cdot \mathbf{n}_i - 1)^2, \quad (9)$$

380 where $|F|$ denotes the number of faces in the mesh and index $k \equiv j + 1 \pmod{4}$.

381 *Orthogonality constraints.* As discussed in 4.2, discrete orthogonality of a face i requires diagonals of equal length, leading to

$$E_{orth} = \sum_{i=0}^{|F|-1} (\|\mathbf{v}_{i2} - \mathbf{v}_{i0}\|^2 - \|\mathbf{v}_{i3} - \mathbf{v}_{i1}\|^2)^2. \quad (10)$$

382 *Static equilibrium constraints.* Following up on Section 3, we provide a generalized energy term of static equilibrium with
383 vertical loads on vertices. Each unsupported vertex \mathbf{v}_i in the quad mesh has adjacent vertices that form a set V_i . For each
384 $\mathbf{v}_j \in V_i$, we introduce a force density w_{ij} as in Eq. (2). We assume the vertical load (along z -axis) on vertex \mathbf{v}_i is p_i , and thus
385 force balance yields

$$E_{static} = \sum_{i=0}^{|V|-1} \left(\sum_{j:\mathbf{v}_j \in V_i} w_{ij}(\mathbf{v}_i - \mathbf{v}_j) - \begin{bmatrix} 0 \\ 0 \\ p_i \end{bmatrix} \right)^2, \quad (11)$$

386 where $|V|$ represents the number of unsupported vertices in the quad mesh. In Figure 10 and 11, p_i represents a uniformly
387 distributed area load on vertices considering the barycentric area around vertex \mathbf{v}_i .

388 *Fairness constraints.* The fairness term is based on second order differences of mesh polylines and the graph Laplacian. For a
389 vertex \mathbf{v}_i which is not on the boundary, its immediate neighbours are $\mathbf{v}_{i0}, \mathbf{v}_{i1}, \dots, \mathbf{v}_{i,n-1}$. The fairness energy of the mesh is
390 calculated by Eq. (12) for regular vertices (valence equal to 4) and Eq. (13) for irregular vertices (valence $n \neq 4$):

$$E_{fairness} = \sum_{i=0}^{|V|-1} \left(\mathbf{v}_i - \frac{\mathbf{v}_{i0} + \mathbf{v}_{i2}}{2} \right)^2 + \left(\mathbf{v}_i - \frac{\mathbf{v}_{i1} + \mathbf{v}_{i3}}{2} \right)^2, \quad (12)$$

$$E_{fairness} = \sum_{i:\mathbf{v}_i \in singularities} \left(\mathbf{v}_i - \frac{\mathbf{v}_{i0} + \dots + \mathbf{v}_{i,n-1}}{n} \right)^2. \quad (13)$$

391 *Proximity constraints.* In the mesh approximation tasks, proximity terms are applied to constrain the mesh vertices to be
392 close enough to the target shape. We use two types of proximity constraints in this paper, expressing squared distances to
393 closest points and to tangent planes at closest points, respectively. The latter has more weight and is known to be a good
394 approximation of the squared distance function of a surface at points near that surface [26]. Given a vertex \mathbf{v}_i , we use a KD-Tree
395 based searching method to find its closest point $\mathbf{v}_{i,ref}$ on the reference surface (represented by a dense triangle mesh). Then
396 the proximity term is calculated by Eq. (14) for point distance or/and by Eq. (15) for tangent distance, in which we introduce
397 the normal vector $\mathbf{n}_{i,ref}$ at $\mathbf{v}_{i,ref}$ on the reference surface,

$$E_{prox} = \sum_{i=0}^{|V|-1} (\mathbf{v}_i - \mathbf{v}_{i,ref})^2, \quad (14)$$

$$E_{prox_tan} = \sum_{i=0}^{|V|-1} ((\mathbf{v}_i - \mathbf{v}_{i,ref}) \cdot \mathbf{n}_{i,ref})^2. \quad (15)$$

398 5.2. Optimization parameters

399 With the constraints mentioned above, we arrive at an objective function E_{total} :

$$E_{total} = \lambda_1 E_{PP} + \lambda_2 E_{PQ} + \lambda_3 E_{angle} + \lambda_4 E_{static} + \lambda_5 E_{fairness} + \lambda_6 E_{prox} + \lambda_7 E_{prox_tan}, \quad (16)$$

400 where λ_5 should be lower than other parameters in an order of magnitude as a “soft” constraint. We solve this optimization
401 problem by a Levenberg-Marquardt method with a fixed damping parameter 10^{-6} . The specific parameters we used for selected
402 figures are listed in Table 2.

403 It is useful to mention such details: Within the iterations, the energy terms of the constraints that need to be fulfilled such
 404 as E_{PP} , E_{PQ} , E_{angle} and E_{static} can be optimized to a value lower than 10^{-3} (the meshes are standardized to the scale with
 405 an average edge length equal to 1). We further check the planarity δ_f of faces by the distance of diagonals divided by average
 406 length of diagonals, which has a typical value lower than 10^{-4} . Similarly, we check the planarity δ_P of polyline P which is the
 407 maximum distance of P 's vertices from a best approximating plane, divided by the bounding box diameter of P . The typical
 408 value of δ_P is lower than 10^{-3} .

Fig.	$ V $	$ F $	#var	#cons	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	#iterations	time[s]/it
6(b)	881	800	5202	13284	1	1	0	0	0.15	0	0.3	10	0.89
6(d)	881	800	5202	12403	1	5	0	0	0.1	0.1	0	10	0.85
10(d)	557	503	4425	7989	1	1	0	1	0.02	0.1	0.2	10	0.63
11(d)	728	655	3803	9351	1	0	0	1	0.1	0.2	0.3	10	0.74
15(a)	350	306	1185	2836	1	0	1	0	0.15	0	0	30	0.49
15(b)	350	306	1185	2836	1	0	1	0	0.1	0	0	10	0.37
15(c)	350	306	1185	2856	1	0	1	0	0.1	0	0	10	0.38
15(d)	350	306	1185	2836	1	0	1	0	0.1	0	0	10	0.45
15(e)	350	306	1185	2836	1	0	0.1	0	0.1	0	0	10	0.37
17(b)	600	539	1986	4967	1	0	1	0	0.03	0	0	10	0.73
17(c)	1071	1020	3429	9282	1	0	1	0	0.03	0	0	10	0.88
17(e)	588	498	2163	4698	1	0	1	0	0.03	0	0	10	0.62
17(g)	640	585	2088	5401	1	0	1	0	0.15	0	0	10	0.39
18(a)	1225	1156	7248	13970	1	1	0	0	0.03	0	0	10	0.92
18(b)	331	288	1923	3531	1	1	0	0	0.05	0	0	10	0.45
19	6241	6084	37212	98123	1	1	0	0	0.1	0.1	0	10	6.51
20	2100	2000	12360	24520	1	1	0	0	0.1	0	0.1	10	1.82

Table 2: This table gives an overview of the size of optimization problems solved for various examples in this paper. We also provide the parameter settings and computation time in seconds. The algorithms are implemented in Python and tested on the CPU processor with Intel Xeon CPU E5-2697 2.60GHz.

409 6. Extended results and discussion

410 6.1. $P+PQ$ meshes

411 In subsection 2.2, we discussed surface approximation by $P+PQ$ meshes initialized with vertical planar beams. Here, we
 412 illustrate more broadly the freedom in the design of $P+PQ$ meshes.

413 Fig. 18 (a) shows an optimization result of a $P+PQ$ mesh with vertical planar polylines. However, when the topology of the
 414 mesh is not regular, it will be no longer possible to keep all the polylines in vertical planes. In this case, we could still achieve
 415 planes intersecting the ground at angles close to 90 degrees (Fig. 18 (b)).

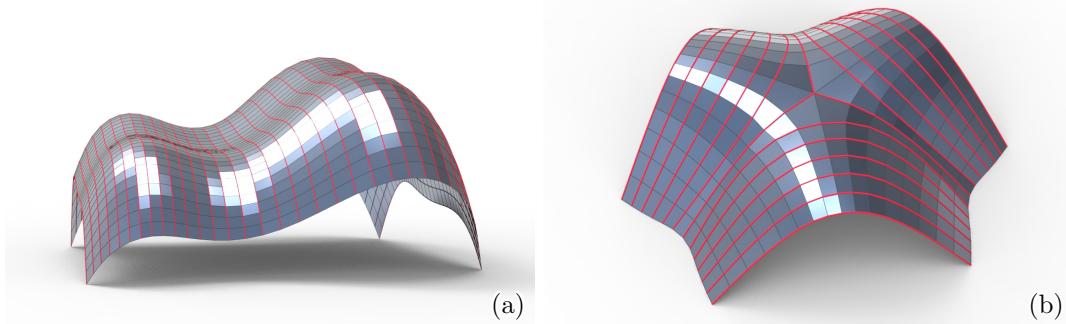


Figure 18: Design of $P+PQ$ meshes. (a) A $P+PQ$ mesh with vertical planar polylines. The vertical planar polylines are colored in red. (b) A $P+PQ$ mesh designed with a combinatorial singularity. Only the planar polylines emanating from the singularity are vertical, others are on nearly vertical planes.

416 Furthermore, we discuss an example for a targeted choice of plane directions for the planar polylines. The planes can be
 417 chosen in order to control the amount of sunlight passing through the support structure. In Fig. 19, we illustrate shading
 418 patterns at different times in a day. There, the application is not shading, but we let enough light pass through the structure.

419 This is done by choosing a sequence of planes for the planar polylines in which each plane is parallel to the incoming light at
 420 different times of the day. To achieve maximal shading, one needs to position the planes as orthogonal as possible to the sun
 421 light. For a discussion of this and related aspects, we refer to Wang et al. [44]. However, that paper does not have planar
 422 parameter lines and thus the fabrication of the support structure would be more complicated.

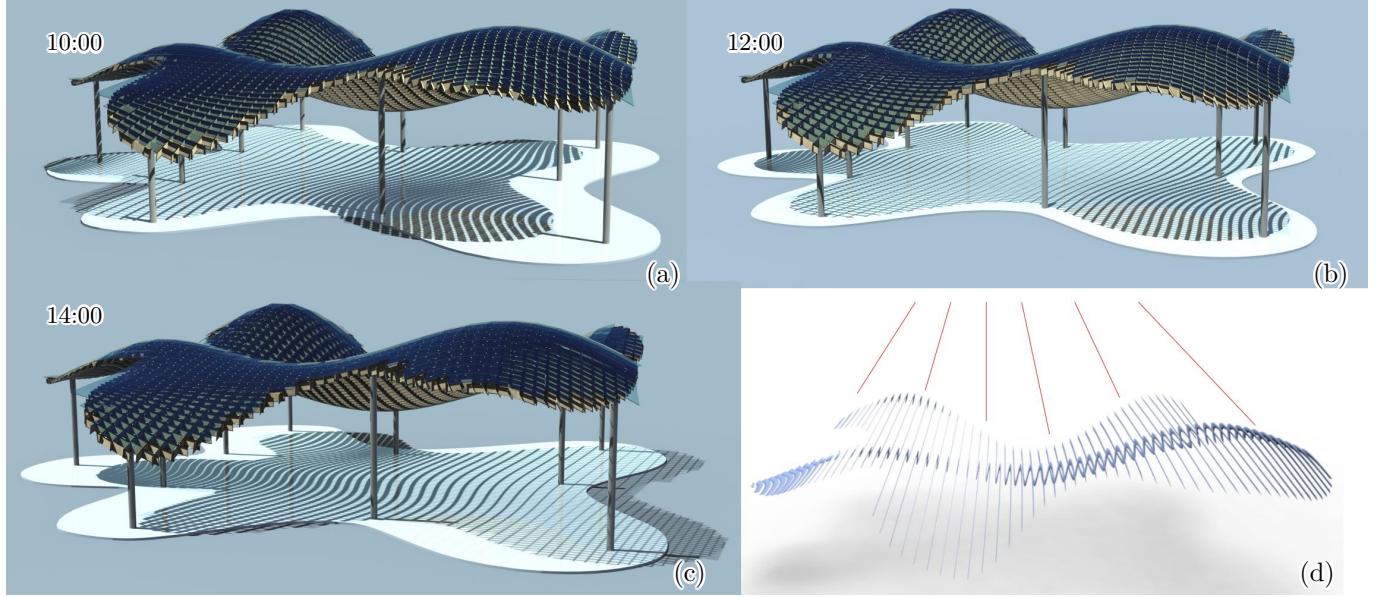


Figure 19: A P+PQ mesh with its support structure generates different shading patterns during a day. (d) illustrates the front view of the support structure aligned with the family of planar polylines. The red lines indicate selected plane positions. When the light is parallel to those, the respective parts in the shading pattern are lighter, well seen in (a-c).

423 In order to achieve more design freedom, we can consider meshes where every N -th polyline in a family is planar. One will
 424 then place the main load-carrying beams of the structure in those planes. As for the Strasbourg train station (Fig. 3 (c)), these
 425 planar beams are the primary load-bearing structures reinforced by the “bike-wheel” cables. Fig. 20 shows the design of a PQ
 426 mesh in which every fifth polyline in one family is planar.

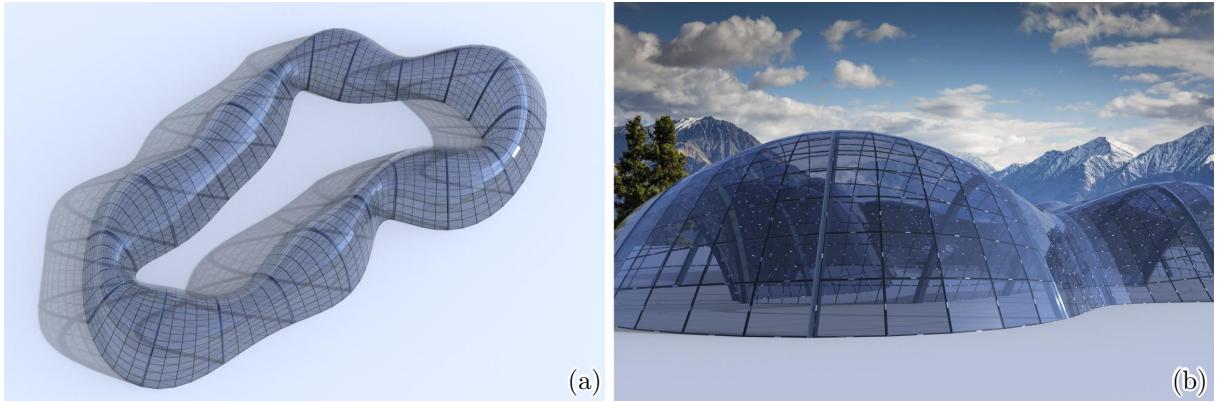


Figure 20: A PQ mesh in which every fifth polyline of a family is planar. Along these polylinies the main load-carrying and stabilizing beams of the structure are placed.

427 6.2. PP+90° meshes

428 In Section 4, we introduced two initialization methods for the design of PP+90° meshes, whose shapes can be further
 429 edited by optimization. Fig. 21 shows more architectural designs of PP+90° meshes initialized by Wunderlich surfaces with a
 430 prescribed base curve on the ground. One can observe that for such combination of constraints, we ask for the orthogonality
 431 of the crossing angles of planar polylines rather than the orthogonality of the planes of the support structure. An important
 432 advantage lies in repetitive nodes. These are not directly present in our discrete versions. However, it is straightforward to
 433 represent each planar parameter line by a smooth curve, e.g. a C^1 cubic spline, so that the arising curve network has precise
 434 orthogonal node angles (see Fig. 22).

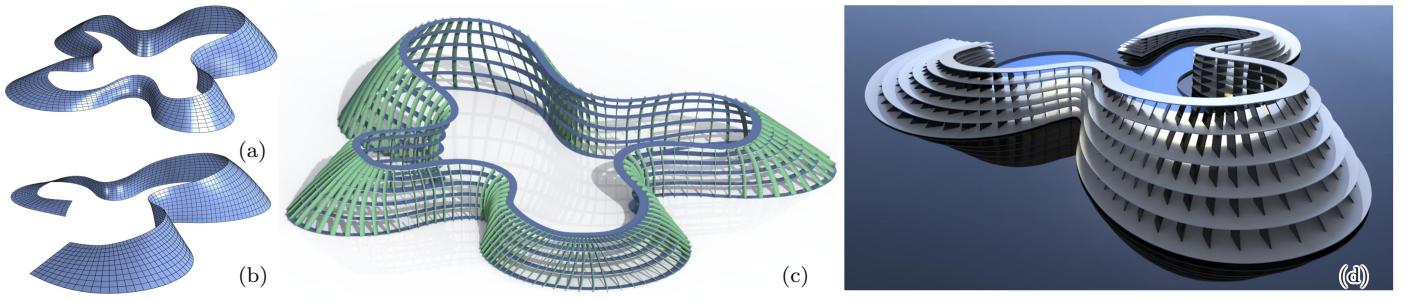


Figure 21: Two PP+90° meshes (a),(b) and corresponding renderings highlighting the planes of supporting structures (c),(d).

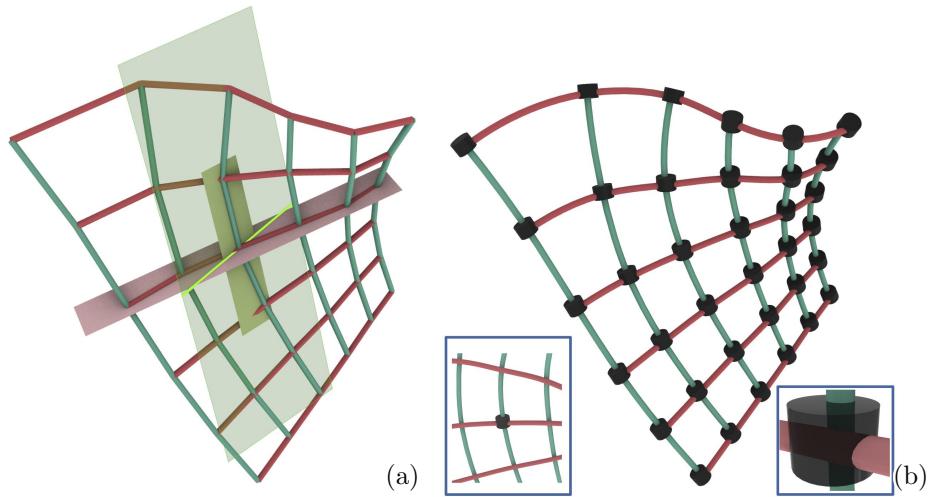


Figure 22: Right node angle and repetitive nodes. (a) A free-form PP+90° mesh has supporting planes which are in general not orthogonal, and the edges do not meet in the nodes at precise right angles. (b) The mesh can be turned into a network of planar smooth curves with precise 90° crossing angles at nodes. One can fabricate it as a network of long-range continuous beams, where one family is stacked on top of the other and fixed by a repetitive node with a right node angle.

435 It is important to mention that accessing the design space through a proper choice of initial shapes is important. Fig. 23
 436 shows an example. Looking for a PP+90° mesh, one may have the idea to start with a PP mesh approximating the desired shape
 437 and then optimize for orthogonality, maybe even closeness to the original shape. The figure illustrates that such an approach
 438 is likely to fail. In general, the more constrained a mesh is, the more care has to be taken in choosing initial shapes. Ideally,
 439 one accesses the design space through shapes which already fulfil all constraints (Wunderlich surfaces in our case) and then
 440 navigates in that space by editing operations.

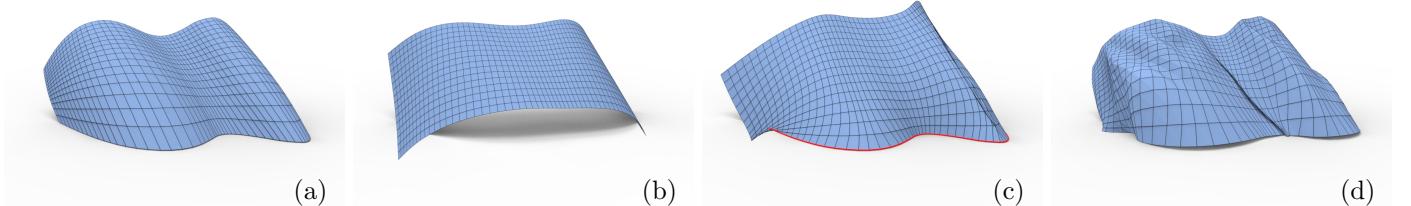


Figure 23: A straightforward approach to PP+90° meshes via optimization based on initialization with a non-orthogonal PP mesh is problematic. (a) A general PP-mesh as initialization. (b) Directly optimizing for right node angles while preserving planarity generates a trivial result with a big shape change. (c) Optimization with a fixed boundary (red) on the ground fails in searching for useful plane directions. (d) Optimization with a closeness constraint to the original shape results in bad fairness.

441 6.3. Feasible constraints

442 To sum up, we discuss the feasible combinations of constraints in quad meshes with planar parameter lines. These combi-
 443 nations are classified into two tasks, approximation or form-finding, according to their shape restrictions (See Table 3). The

444 former one can at least locally approximate a given surface but the latter one is restricted to a form-finding task. Here we
 445 point out that the approximation task with $P+90^\circ$ meshes can be easily achieved by prescribing a family of planes which
 446 contain the planar polylines and remeshing on the corresponding orthogonal fields, taking subsection 2.2 as a reference. For the
 447 form-finding task of $PP+F$ meshes, one can take a $P+F$ mesh and its respective force density as initialization, then it would
 448 be possible to find $PP+F$ meshes with shape deviation.

Combination of constraints	P-mesh	PP-mesh
Planar Quad (PQ)	Approximation	Form-finding
Orthogonal parameter lines (90°)	Approximation	Form-finding
Principal mesh ($PQ+90^\circ$)	Form-finding	Form-finding
Static Equilibrium (F)	Approximation	Form-finding

Table 3: An overview of the constraint combinations in quad meshes with planar polylines. The constraint combinations are either discussed in this paper (red) or in [13] (blue).

449 7. Conclusion and future research

450 In this article, we investigated the design space of meshes with planar polylines in the presence of additional constraints.
 451 We provided a solution for approximating a given surface by a $P+PQ$ mesh. In addition, we introduced two initialization
 452 methods for $PP+90^\circ$ meshes, and used optimization-based editing to explore the shape space. Also, we considered meshes
 453 with planar polylines in static equilibrium, without vertical load from the geometric perspective, as well as with vertical load
 454 for the architectural rationalization. long-range planar beams provide clear advantages for manufacturing, but appear to have
 455 structural benefits as well. Finally, we provided details on our computational framework and illustrated its capabilities by a
 456 number of examples.

457 Directions for future research include automatic solutions for the approximation problem with $P+PQ$ meshes, especially
 458 when a global solution has to use piecewise planar parameter lines. We also did not yet provide a general strategy for the
 459 design of $P+PQ+F$ meshes. In our paper, Wunderlich surfaces turned out to be very helpful for the design of $PP+90^\circ$ meshes.
 460 Therefore, one would expect great benefits from explicit solutions of the PDE (7). We have already made some progress in this
 461 direction and hope to present sufficiently large and practically useful classes of surfaces with an orthogonal family of planar
 462 parameter lines in the near future.

463 Acknowledgements

464 This research has been supported by KAUST baseline funding (grant BAS/1/1679-01-01). We are very grateful to Davide
 465 Pellis and Yu-Chou Chiang for sharing Airy stress surfaces and valuable discussion on static equilibrium.

466 References

- [1] Sigrid Adriaenssens, M Asce, Laurent Ney, Eric Bodarwe, and Chris J.K. Williams, *Finding the Form of an Irregular Meshed Steel and Glass Shell Based on Construction Constraints*, Journal of Architectural Engineering **18** (2012), no. September, 206–213.
- [2] Niccolo Baldassini, Helmut Pottmann, Jacques Raynaud, and Alexander Schiftner, *New Strategies and Developments in Transparent Free-Form Design : From Faceted to Nearly Smooth Envelopes*, International Journal of Space Structures **25** (2010), no. 3, 185–197.
- [3] Bernd Bickel, Paolo Cignoni, Luigi Malomo, and Nico Pietroni, *State of the art on stylized fabrication*, Computer Graphics Forum **37** (2018), no. 6, 325–342.
- [4] Wilhelm Blaschke, *Vorlesungen über Differentialgeometrie*, vol. 2, Springer, 1923.
- [5] Alexander I. Bobenko and Yuri B. Suris, *Discrete differential geometry. integrable structure*, Graduate Studies in Mathematics, vol. 98, American Mathematical Society, Providence, RI, 2008.
- [6] David Bommes, Henrik Zimmer, and Leif Kobbelt, *Mixed-integer quadrangulation*, ACM Trans. Graph. **28** (2009), no. 3.
- [7] Frederic Cazals and Marc Pouget, *Estimating differential quantities using polynomial fitting of osculating jets*, vol. 22, 05 2003, pp. 177–187.
- [8] Yu-Chou Chiang, *Design and fabrication of shell structures: Aided by radial basis functions and reconfigurable mechanisms*, Ph.D. thesis, Delft University of Technology, 2022.
- [9] Bailin Deng, Helmut Pottmann, and Johannes Wallner, *Functional webs for freeform architecture*, Computer Graphics Forum **30** (2011), 1369–1378, Proc. Symp. Geometry Processing.
- [10] Manfredo do Carmo, *Differential geometry of curves and surfaces*, Prentice-Hall, 1976.
- [11] Thomas Ivey, *Surfaces with orthogonal families of circles*, Proc. Amer. Math. Soc. **123** (1995), 865–872.
- [12] Alec Jacobson, Daniele Panozzo, et al., *libigl: A simple C++ geometry processing library*, 2018, <https://libigl.github.io/>.
- [13] Caigui Jiang, Cheng Wang, Xavier Tellier, Johannes Wallner, and Helmut Pottmann, *Planar panels and supporting beams in architectural structures*, (2021), Manuscript submitted for publication.
- [14] Yang Liu, Helmut Pottmann, Johannes Wallner, Yong-Liang Yang, and Wenping Wang, *Geometric modeling with conical meshes and developable surfaces*, ACM Trans. Graphics **25** (2006), no. 3, 681–689.
- [15] Koliann Mam, *Exploration structurelle et environnementale des ouvrages en bois de grande portée*, Phd thesis, Ecole des Ponts ParisTech, 2021.
- [16] James Clerk Maxwell, *On reciprocal figures, frames, and diagrams of forces*, Trans. R. Soc. Edinburgh **26** (1872), 1–40.

- [491] [17] Shane McCormick, Charles Besjak, D. Stanton Korista, and William Baker, *Schubert club band shell, saint paul, minnesota, usa*, Structural Engineering International **14** (2004), no. 2, 140–141.
- [492] [18] James McCrae, Karan Singh, and Niloy J. Mitra, *Slices: A shape-proxy based on planar sections*, ACM Trans. Graph. **30** (2011), no. 6, 1–12.
- [493] [19] Romain Mesnil, Cyril Douthe, Olivier Baverel, and Bruno Léger, *Marionette meshes: modelling free-form architecture with planar facets*, International Journal of Space Structures **32** (2017), no. 3-4, 184–198.
- [494] [20] Romain Mesnil, Cyril Douthe, Olivier Baverel, and Bruno Léger, *Morphogenesis of surfaces with planar lines of curvature and application to architectural design*, Automation in Construction **95** (2018), 129–141.
- [495] [21] Cameron Millar, Toby Mitchell, Arek Mazurek, Ashpica Chhabra, Alessandro Beghini, Jeanne N. Clelland, Allan McRobie, and William F. Baker, *On designing plane-faced funicular gridshells*, Manuscript submitted for publication, 2021.
- [496] [22] Cameron Millar, Toby Mitchell, Arek Mazurek, Ashpica Chhabra, Alessandro Beghini, Allan McRobie, and William Baker, *On funicular gridshells and Airy stress functions*, Proceedings of the IASS Symposium 2020/21, 2021.
- [497] [23] Davide Pellis and Helmut Pottmann, *Aligning principal stress and curvature directions*, Advances in Architectural Geometry, 2018.
- [498] [24] Davide Pellis, *Quad meshes as optimized architectural freeform structures*, Ph.D. thesis, Wien, 2019.
- [499] [25] Helmut Pottmann, Michael Eigensatz, Amir Vaxman, and Johannes Wallner, *Architectural geometry*, Computers and Graphics **47** (2015), 145–164.
- [500] [26] Helmut Pottmann and Michael Hofer, *Geometry of the squared distance function to curves and surfaces*, Visualization and Mathematics III (H.-C. Hege and K. Polthier, eds.), Springer, 2003, pp. 223–244. MR 2005a:53004
- [501] [27] Helmut Pottmann and Yang Liu, *Discrete surfaces in isotropic geometry*, Mathematics of Surfaces XII (Berlin, Heidelberg) (Ralph Martin, Malcolm Sabin, and Joab Winkler, eds.), Springer Berlin Heidelberg, 2007, pp. 341–363.
- [502] [28] Helmut Pottmann, Yang Liu, Johannes Wallner, Alexander Bobenko, and Wenping Wang, *Geometry of multi-layer freeform structures for architecture*, ACM Trans. Graph. **26** (2007), no. 3, 65–es.
- [503] [29] Helmut Pottmann and Johannes Wallner, *Computational line geometry*, Springer, 2001.
- [504] [30] Hans Sachs, *Die Strahlflächen mit durchwegs ebenen Orthogonaltrajektorien der Erzeugenden*, Arch. Math. **21** (1970), 437–444.
- [505] [31] Robert Sauer, *Differenzengeometrie*, Springer, 1970.
- [506] [32] Jörg Schlaich and Hans Schober, *Glass-Covered Grid-Shells*, Structural Engineer (1996), 88–90.
- [507] [33] Volker Schmid, Jan Peter Koppitz, and Anja Thurik, *Neue Konzepte im Holzbau mit Furnierschichtholz - Die Holztragkonstruktion des Metropol Parasol in Sevilla*, Bautechnik **88** (2011), no. 10, 707–714.
- [508] [34] Hans Schober, *Transparent Shells: Form, Topology, Structure*, Ernst & Sohn, 2015.
- [509] [35] Yuliy Schwartzburg and Mark Pauly, *Fabrication-aware design with intersecting planar pieces*, Wiley Online Library (2013).
- [510] [36] John Sharp, *Sliceforms*, Norfolk, UK: Tarquín Publications (1999).
- [511] [37] Werner Sobek and Lucio Blandini, *The Mansueto Library - Notes on a glazed steel grid shell from design to construction*, Challenging Glass 2 - Conference on Architectural and Structural Applications of Glass, CGC 2010 (2010), no. May, 179–185.
- [512] [38] Chengcheng Tang, Xiang Sun, Alexandra Gomes, Johannes Wallner, and Helmut Pottmann, *Form-finding with polyhedral meshes made simple*, ACM Trans. Graph. **33** (2014), no. 4, 70:1–70:9.
- [513] [39] Xavier Tellier, *Morphogenesis of curved structural envelopes under fabrication constraints*, Ph.D. thesis, Univ. Paris-Est, 2020.
- [514] [40] Xavier Tellier, Cyril Douthe, Laurent Hauswirth, and Olivier Baverel, *Surfaces with planar curvature lines: Discretization, generation and application to the rationalization of curved architectural envelopes*, Automation in Construction **106** (2019).
- [515] [41] ———, *Funicularity of conics*, Acta Mechanica **232** (2021), no. 8, 3179–3191.
- [516] [42] Hans Trautwein, *Neue Flächen mit ebenen Falllinien*, Ph.D. thesis, TU Munich, 1996.
- [517] [43] Etienne Vouga, Mathias Höbinger, Johannes Wallner, and Helmut Pottmann, *Design of self-supporting surfaces*, ACM Trans. Graph. **31** (2012), no. 4.
- [518] [44] J. Wang, C. Jiang, P. Bompas, J. Wallner, and H. Pottmann, *Discrete line congruences for shading and lighting*, Computer Graphics Forum **32** (2013), no. 5, 53–62.
- [519] [45] Walter Wunderlich, *Flächen mit ebenen Falllinien*, Monatsh. Math. **65** (1961), 291–300.
- [520] [46] ———, *Flächen mit Kegelschnitten als Falllinien*, J. Reine Angew. Math. **208** (1961), 204–220.
- [521] [47] ———, *Betrag- und Potentialflächen mit ebenen Falllinien*, Sitzungsber. Österr. Akad. Wiss., Math.-Naturw. Kl. **170** (1962), 105–120.
- [522] [48] ———, *Römerflächen mit ebenen Falllinien*, Annali di Matematica pura ad applicata **57** (1962), 97–108.