EUROGRAPHICS 2015 / O. Sorkine-Hornung and M. Wimmer
(Guest Editors)

Volume 34 (2015), Number 2

Statics Aware Grid Shells

Nico Pietroni[1†], Davide Tonelli[2], Enrico Puppo[3], Maurizio Froli[2], Roberto Scopigno[1] and Paolo Cignoni[1]

[1]ISTI, CNR, Italy [2]University of Pisa [3]University of Genova

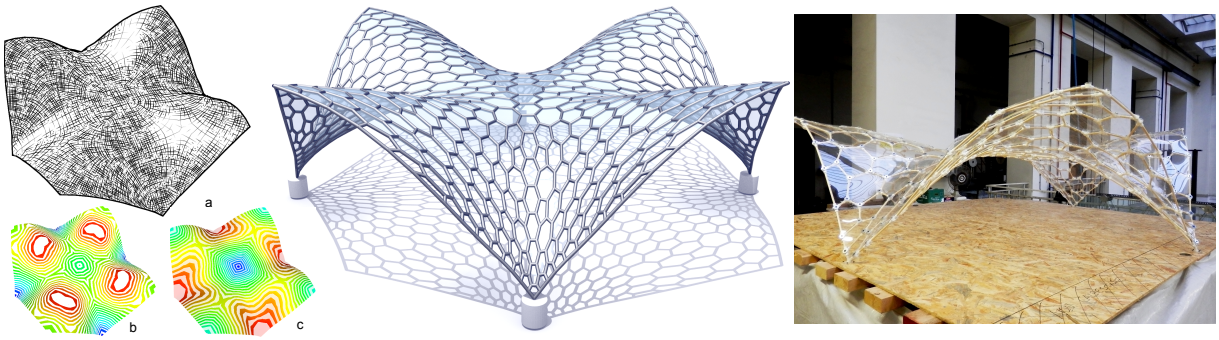


Figure 1: We perform a FEM static analysis of the input surface to obtain a stress field, from which we derive a double orthogonal line field (a), an anisotropy scalar field (b) and a density scalar field (c). Then we build a polygonal tessellation whose elements are sized and aligned according to the stress tensor field; this tessellation is optimized for symmetry and regularity of faces. A small scale model is fabricated to validate the result with load tests.

Abstract

We introduce a framework for the generation of polygonal grid-shell architectural structures, whose topology is designed in order to excel in static performances. We start from the analysis of stress on the input surface and we use the resulting tensor field to induce an anisotropic non-Euclidean metric over it. This metric is derived by studying the relation between the stress tensor over a continuous shell and the optimal shape of polygons in a corresponding grid-shell. Polygonal meshes with uniform density and isotropic cells under this metric exhibit variable density and anisotropy in Euclidean space, thus achieving a better distribution of the strain energy over their elements. Meshes are further optimized taking into account symmetry and regularity of cells to improve aesthetics. We experiment with quad meshes and hex-dominant meshes, demonstrating that our grid-shells achieve better static performances than state-of-the-art grid-shells.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Grid-shells, such as steel-glass structures, have been used for about forty years in architecture [OR95]. While triangle-based grid-shells seem unbeatable from the point of view of strength, quad-based structures have become popular in the last decade, because of their improved aesthetics and nice mathematical properties. According to recent trends, also hex-dominant structures exhibit interesting geometric and static properties [PJH*14, JWWP14].

Most of the design methods proposed in the literature are concerned with the optimization of grid-shells by acting on their geometry: the grid topology is given in input, while the shape, as conceived by the architect, can be modified in order to improve either the geometric properties, or the static performance of the grid-shell. Conversely, in this work we

† e-mail: nico.pietroni@isti.cnr.it

demonstrate how it is possible to improve the static performance of a grid-shell by changing its tessellation only, while keeping the original shape fixed.

The intuition behind our approach stems from the observation of light and strong lattice structures that appear in nature, such as bones. The basic idea is to concentrate more supporting material where the structure undergoes more stress; this principle has also been applied recently to 3D printing [LSZ*14]. Besides, the stiffness of a grid-shell structure can be improved by aligning its elements to the principal stress directions of the underlying surface, thus obtaining an anisotropic grid [TG69, TWK87].

A rigorous development of such intuition is hard, though, because the static behavior of a grid-shell may change dramatically with its connectivity, while the problem of finding the best grid topology for a given shape has a combinatorial nature. In order to bring the problem to a tractable form, we study the statics of a *continuous shell* representing the input surface, and we use the resulting stress field to set the density and anisotropy of cells in our *grid-shell*.

In this scenario our main contributions can be summarized as follows:

- We define the relation between optimal anisotropy and density of a grid-shell structure with respect to a given stress field computed on a corresponding continuous shell.
- We propose a new method to improve the static performance of a grid-shell structure by optimizing its tessellation only, without changing the surface shape. The process is illustrated in figure 1: we first perform a static analysis of the surface, from which we obtain an anisotropic, non-Euclidean metric induced by a double orthogonal line field, a density field and an anisotropy field. Next, we build an anisotropic grid-shell driven by such metric.
- We apply geometric optimization to follow surface symmetries and to improve the local shape of faces of our mesh, making them closer to the faces of Archimedeal solids; this geometric optimization phase greatly contributes to improve the aesthetics of our grid-shells, and it also slightly improves static performances.
- We demonstrate the effectiveness of the proposed method when applied to quadrilateral and hex-dominant meshing, by comparing it to state-of-the-art models. Results are validated by performing static FEM analysis and by using a reduced scale fabricated model on which we performed load tests.

2. Related Work

2.1. Statics of grid-shell structures

Grid-shell structures are a modern response to the ancient need of covering long span spaces. Their supporting structure is made of *beams* connected at *joints*, while covering panels only act as load. The load bearing capacity of a grid-shell is directly related to the connectivity of its corresponding mesh. While triangular meshes are more rigid and stronger than any other competitor, polygonal meshes have some advantages in terms of ease of construction and lend themselves to the design of torsion-free structures [PJH*14]. Generally speaking, beams and joints are subject to axial forces as well as bending moments. It is well known [BK01] that the form of a polygonal mesh is maintained only if the joints are able to develop bending moments, while triangular meshes maintain their form even if the joints are hinges.

In *compressive structures*, the principal stress comes mainly from axial forces, and this explains the deep interconnection between them and masonry structures. A purely compressive grid-shell can be obtained only through a form-finding process, aimed at finding the *funicular surface* that fits the given boundary constraints [BK01, OKF08, OR95]. In this case, an initial form, as designed by an architect, is taken just as a guide to obtain the final structure, which may differ significantly from it. Thrust Network Analysis [Blo09], a recent form-finding method specific to masonry [PBSH13], has also been extended to the computation of triangular grids in static equilibrium [LPS*13]. A further extension of this method introduced recently by Tang et al. [TSG*14] directly allows for grid-shell form finding: not only it computes the target funicular surface, but it also optimizes the positions of edges. In Section 7, we compare some of our results with grid-shells obtained with this latter method.

If the input form is to be maintained, obtaining a funicular structure is generally not possible. Nevertheless, mesh topology should be designed in order to obtain a good static behavior, e.g., by distributing the load as uniformly as possible among the different beams and by reducing the bending moments in both the beams and the joints. Some comparative parametric analyses [MW13] have been carried out about the influence of the remeshing pattern on the grid-shell load bearing capacity. There exist surprisingly few studies about the optimal (in a structural sense) connectivity and distribution of edges [SB10], although probably these are – in conjunction with the surface shape – the most influential parameters that govern the structural behavior of the grid-shell. For our comparative experiments, we adopt the equivalence criterion of simultaneous equal total mass and equal total length of edges, as in [MW13].

2.2. Architectural geometry

Most contributions in this field are concerned with the optimization of geometric properties of polygonal meshes approximating a free-form surface. Many works address the planarity of faces, such the construction of PQ (planar quad) meshes [LPW*06, LXW*11, TSG*14, SB10, YYPM11, ZSW10], CP (circle packing) meshes [SHWP09], and polygonal hex-dominant meshes [CW07, JWWP14, PJH*14, SHWP09, Tro08]. Others try to build meshes from a restricted number of tiles or molds [EKS*10, FLHCO10,

SS10, ZCBK12]. A few works address the realization of support structures, parallel meshes and torsion-free meshes [PLW*07, PJH*14, TSG*14]. Among these works, only few focus on the design of a grid topology [CW07, LXW*11, SB10, ZSW10] and just Schiftner and Balzer [SB10] take into account statics.

3. Statics of grid shells

Robustness of a grid-shell structure is strictly related to the distribution of load among its beams. Intuitively, the more uniform the distribution, the stronger the structure. Moreover, the better beams are aligned to principal stress directions, the more forces act axially on them, while bending moments are reduced. This leads to a structure with better statics properties.

Due to the complex non-linear and combinatorial nature of the problem, finding an optimal connectivity may be a very hard, non feasible task. In fact, even a small change in connectivity may drastically change the distribution of load.

Rather than setting the problem as an optimization, we tackle it in a different way. We first study the statics of a continuous shell corresponding to the input surface; this provides us a *stress field* encoding how tensions propagate through the surface. We use this stress field to infer preferred density and anisotropy of a grid-shell covering the same surface. To this aim, we study the relations between an ideal regular grid-shell subject to a constant stress field, in order to understand the laws that relate *locally* the density and anisotropy of forces to the size and anisotropy of elements of the mesh. Next, we induce a non-Euclidean metric over the surface, which follows the directions of the stress field, and whose density and anisotropy are set according to the proportionality laws obtained from the local analysis. This metric will be used in Section 4 to compute a corresponding anisotropic meshing. The following subsections provide details on the analysis that lead us to the computation of the anisotropic metric.

3.1. Static analysis of a continuous shell

We first consider the input surface as a continuous elastic thin shell $\mathcal{S}$ subject to uniform projected load and with pin-restrained boundary nodes. In order to perform Finite Element Analysis (FEA), this shell is discretized into structural triangular faces with uniform thickness and material properties. We analyze this surface with the Displacement Method [Bat96], which is a standard in statics analysis. Details are provided in Appendix A.

The FEA returns a tensor field, called the Cauchy stress tensor, which encodes, at each point p of $\mathcal{S}$ and for any possible direction $\vec{t}$, how much force goes through a differential of surface centered at p and orthogonal to $\vec{t}$. In particular, a shell can be treated as a two-dimensional problem in theory of elasticity, which means that tension orthogonal to the tangent plane is negligible and, therefore, the Cauchy stress tensor has rank 2. Its two eigenvectors and eigenvalues represent the principal directions and the principal stresses at each point, respectively.

The principal directions and stresses are treated as a double orthogonal line field $\Psi_f(p) = (\vec{u_f}(p), \vec{v_f}(p))$ where $\vec{u_f}$ and $\vec{v_f}$ lie on the tangent bundle of $\mathcal{S}$ and define the minimum and maximum stress at each point p of the surface, respectively.

3.2. From continuous shells to grid-shells

Our grid-shell is structurally different from the continuous shell on which we performed FEA. A transition from the continuum shell to a discrete *but "equivalent"* grid-shell must be set up, but this is neither trivial nor straightforward. Specifically, a suitable "transfer" criterion must be devised in order to make this relation clear and consistent. In the following, we show how *topology-specific "transfer" criteria* can be conveniently built. This is done by relating the local stress tensor Ψ_f with a tailored metric tensor Ψ_e (to be used for remeshing purpose), by means of a relevant integral-based function: the strain energy $\mathbf{U}$.

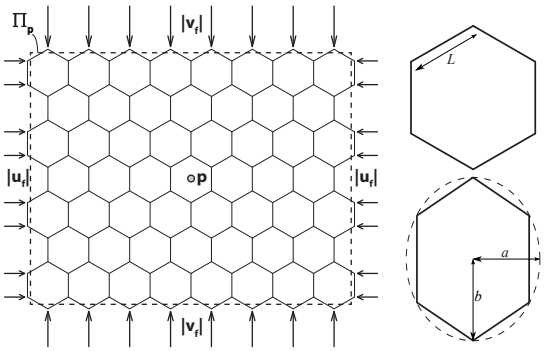
3.2.1. Problem statement

Consider a neighbourhood Π_p of a point $p \in \mathcal{S}$, subject to a field of tensions $\Psi_f(p)$, and suppose to fit here a topologically regular (but generally non isotropic) hexagonal grid aligned with the principal directions $(\vec{u_f}, \vec{v_f})$, as shown in Figure 2. Besides suppose, just for the sake of clarity, that $\vec{u_f}$ and $\vec{v_f}$ are the horizontal and vertical directions respectively. Then, the norms $(|\vec{u_f}|, |\vec{v_f}|)$ can be regarded as integrals of evenly distributed stresses acting along the horizontal and vertical boundaries of the neighbourhood Π_p, respectively. Under the action of these forces we assume that the hexagonal grid can shrink or stretch out, both isotropically or anisotropically, according to the 'transfer' criterion defined in the following and provided that each hexagon keeps inscribed within an ellipse of radii a and b measured along $\vec{u_f}$ and $\vec{v_f}$ (see Figure 2 bottom right), respectively. Within this framework, the strain energy $\mathbf{U}(\vec{u_f}, \vec{v_f}, a, b)$ is used to put into relation the forces $(|\vec{u_f}|, |\vec{v_f}|)$ of the continuous shell (i.e. the statics quantities) with the lengths (a, b) of the corresponding grid-shell cells (i.e., the metric quantities).

We further define $L = \sqrt{ab}$ as the *scale* of the ellipse. In the isotropic case, we have $|\vec{u}| = |\vec{v}|$ and $L = a = b$: in this case the expression of the strain energy can be simplified to the *density strain energy* $\mathbf{U}_d(|\vec{u}|, L)$. In the anisotropic case, if we fix the scale of both forces and lengths, we can define an *anisotropy strain energy* $\mathbf{U}_a(\frac{\vec{v_f}}{\vec{u_f}}, \frac{b}{a})$.

3.2.2. Transfer criterion: a tailored non-Euclidean metric

We introduce a new non-Euclidean metric g_Ψ over the surface $\mathcal{S}$, that allows us to compute optimal grid-shell cells.



(a) Neighbourhood Π_p of a point $p \in \mathcal{S}$. (b) Characterization of the hex-grid shape in Π_p.

Figure 2: Problem statement. Local fitting of a regular hexagonal grid in a neighbourhood Π_p of a point $p \in \mathcal{S}$, aligned with the principal directions $\vec{u_f}, \vec{v_f}$ and subject to boundary forces $|\vec{u_f}|, |\vec{v_f}|$ (left). Geometric dimensions of isotropic (top right) and anisotropic cells (bottom right).

Specifically, this metric maps each unit circle on the tangent plane centered at $p \in \mathcal{S}$ to an optimal ellipse of radii (a,b), as shown in Figure 3. For given forces $(\vec{v_f}, \vec{u_f})$ we want to set L in the isotropic case, and b/a in the anisotropic case, such that the density strain energy $\mathbf{U}_d(|\vec{u}|, L)$ is kept constant over the entire surface $\mathcal{S}$, while the anisotropy strain energy $\mathbf{U}_a(\frac{\vec{v_f}}{\vec{u_f}}, \frac{b}{a})$ is minimized for the local stress ratio $\frac{|\vec{v_f}|}{|\vec{u_f}|}$.

In order to set up g_Ψ, we define a new metric tensor $\Psi_e = (\vec{u_e}, \vec{v_e})$, which is a rescaled version of Ψ_f. We introduce a pair of parameters (d_e, a_e) called *density* and *anisotropy* of the tensor Ψ_e, respectively, defined as follows:

$$\begin{cases} d_e = |u_e||v_e| & (1) \\ a_e = \dfrac{|u_e|}{|v_e|} & (2) \end{cases}$$

Note that only the directions and sizes of $(\vec{u_e}, \vec{v_e})$ are relevant to Ψ_e, not their orientations. Therefore, since $\vec{u_e}$ and $\vec{v_e}$ are orthogonal, we decouple the scalar and directional information and represent Ψ_e as a triple $(\vec{u}_n, d_e, a_e)$, where $\vec{u}_n$ is a unit-length vector parallel to maximal stress and d_e and a_e are the density the anisotropy as defined above. The metric induced by Ψ_e on $\mathcal{S}$ is given by the symmetric tensor $g_\Psi = \mathbf{W}^T\mathbf{W}$, with

$$\mathbf{W} = \begin{bmatrix} \sqrt{d_e a_e} & 0 \\ 0 & \sqrt{d_e / a_e} \end{bmatrix} \quad (3)$$

where matrix $\mathbf{W}$ is expressed at each point in a local coordinate system having its first axis aligned with $\vec{u}_n$. The metric g_Ψ is hence completely defined by the two scalar parameters (d_e, a_e), which can be determined by putting in relation the stress tensor Ψ_f with the metric tensor Ψ_e as in the following.

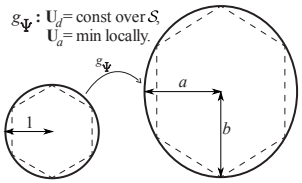


Figure 3: 'Transfer' Criterion: a new metric g_Ψ over $\mathcal{S}$.

3.2.3. Density parameter d_e

Now the focus is on *isotropic* hexagonal polygons (see Figure 2 (top right)) subject to an isotropic state of stress $|\vec{u}|$ (i.e., $|\vec{u_f}| = |\vec{v_f}| = |\vec{u}|$), and the problem is that of associating a suitable size to each grid-shell cell over $\mathcal{S}$, in such a way that all of them share the same constant strain energy $\mathbf{U}_d(|\vec{u}|, L) = \text{const}$. Therefore, the *density* 'transfer' criterion translates into a scalar field that associates a convenient polygon edge length L to each point $p \in \mathcal{S}$. To pin it down, it is mandatory to derive the expression of the density strain energy $\mathbf{U}_d(|\vec{u}|, L)$.

In an isotropic hexagonal grid, as the one in Figure 4(a), each beam is loaded only at the nodes. This in turn means that $\mathbf{U}_d$ assumes the same expression for each beam, hence the analysis can be contained to a single beam. For a planar beam of length L, subject to in-plane loads only, the strain energy $\mathbf{U}$ is given by:

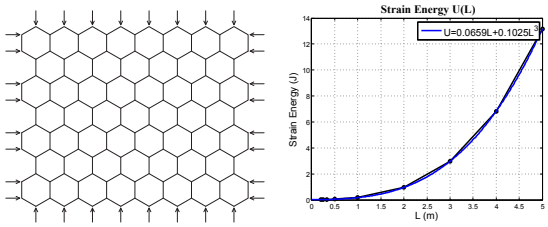
$$\mathbf{U} = \frac{1}{2}\Big[\int_0^L N\varepsilon + V\gamma + M\chi \, dz\Big] \quad (4)$$

where N is the axial force, ε the axial strain, V the shear force, γ the shear strain, M the bending moment, χ the curvature and z the curvilinear abscissa of the beam. Equation (4) can also be expressed in terms of the internal forces only:

$$\mathbf{U} = \frac{1}{2}\Big[\int_0^L \frac{N^2}{EA} + \frac{V^2}{GA_v} + \frac{M^2}{EI} \, dz\Big] \quad (5)$$

where EA, GA_v, EI are the axial stiffness, shear stiffness and bending stiffness of the beam, respectively, whereas A, A_v, I are the cross area, shear area and moment of inertia of the cross section of the beam, respectively.
Additionally, when the beam is loaded only at the nodes by a generic force of magnitude $|\vec{u}|$, the internal forces assume the following form:

$$\begin{cases} N &= \alpha|\vec{u}| \\ V &= \beta|\vec{u}| \\ M &= (\gamma L - \delta z)|\vec{u}| \end{cases} \quad (6)$$



(a) Isotropic hexagonal grid subjected to an *isotropic* constant state of stress.

(b) Plot of the deformation energy as a function of the isotropic grid edge length L.

Figure 4: Derivation of $\mathbf{U}_d(|\vec{u}|,L)$ for an *isotropic* hex-grid subject to an isotropic state of stress.

so that the strain energy becomes:

$$\begin{aligned}\mathbf{U}_d(|\vec{u}|,L) &= \frac{1}{2}\Big[\int_0^L \frac{(\alpha|\vec{u}|)^2}{EA} + \frac{(\beta|\vec{u}|)^2}{GA_v} + \frac{(\gamma L-\delta z)^2(|\vec{u}|)^2}{EI}\,dz\Big] \\ &= \frac{|\vec{u}|^2}{2}L\Big[\frac{\alpha^2}{EA} + \frac{\beta^2}{GA_v} + \frac{(\gamma^2+\frac{\delta^2}{3}-\gamma\delta)}{EI}L^2\Big] \\ &= \frac{|\vec{u}|^2}{2}\big(\varepsilon L+\zeta L^3\big)\end{aligned} \tag{7}$$

It is easily seen that in most structural applications $\zeta > \varepsilon$ so that $\zeta L^3 \gg \varepsilon L$, hence the linear term can be neglected and simply we end up with $\mathbf{U}_d(|\vec{u}|,L) \simeq |\vec{u}|^2 L^3$, that holds for a single beam as well as for the whole grid of Figure 4(a).

Figure 4(b) shows the perfect agreement between equation (7) and a numerical experiment performed on the grid of Figure 4(a), carried out by keeping the force $|\vec{u}|$ constant and letting L vary. It shows clearly that the function $\mathbf{U}_d(|\vec{u}|,L)$ is a cubic polynomial in L, whose coefficients depend only on the beams' cross section (i.e. A and I) and material (i.e Young modulus E). It is also worth noting that the coefficient ε of the linear term is numerically negligible with respect to ζ, and this proves the validity of the approximation $\mathbf{U}_d(|\vec{u}|,L) \propto |\vec{u}|^2 L^3$. With this relationship at hand, and recalling from equation (1) that under isotropic state of stress $d_f = |u|^2$ and $d_e = |u_e|^2$, the following equations hold:

$$\begin{cases} L^3 \propto \dfrac{1}{|u|^2} = \dfrac{1}{d_f} \;\rightarrow L \propto d_f^{-\frac{1}{3}} & (8) \\ L \propto \dfrac{1}{|u_e|} = \dfrac{1}{\sqrt{d_e}} \rightarrow L \propto d_e^{-\frac{1}{2}} & (9) \end{cases}$$

Therefore it follows that the densities d_e and d_f, respectively belonging to the metric tensor Ψ_e and the stress tensor Ψ_f, are related as in the following:

$$d_e = d_f^{\frac{2}{3}} = \big(|u_f||v_f|\big)^{\frac{2}{3}} \tag{10}$$

3.2.4. Anisotropy parameter a_e

Now the focus is on *anisotropic* hexagonal polygons (see Figure 2 (bottom right)) subject to an *anisotropic* state of stress $(|\vec{u_f}|,|\vec{v_f}|)$, and the problem is that of associating a suitable ratio $\frac{b}{a}$ to each grid-shell cell over $\mathcal{S}$, in such a way that the anisotropy strain energy $\mathbf{U}_a(\frac{\vec{v_f}}{\vec{u_f}},\frac{b}{a})$ results locally minimized. Therefore, also the *anisotropy* 'transfer' criterion consists of a scalar field which associates a convenient polygon anisotropy ratio $\frac{b}{a}$ to each point $p \in \mathcal{S}$. To pin it down, it is mandatory to derive the expression for the minimum of the anisotropy strain energy $\min\big[\mathbf{U}_a(\frac{\vec{v_f}}{\vec{u_f}},\frac{b}{a})\big]$, where the sum $|\vec{u_f}|+|\vec{v_f}|$ is kept constant.

While obtaining an analytic result is rather trivial for the density case, things are not easy for the anisotropy. Therefore, the relationship $\min\big[\mathbf{U}_a(\frac{\vec{v_f}}{\vec{u_f}},\frac{b}{a})\big]$ can only be obtained numerically. Figure 5(b) shows the results of a parametric investigation in which the strain energy $\mathbf{U}_a$ has been computed for several couples of ratios $\frac{\vec{v_f}}{\vec{u_f}}$ and $\frac{b}{a}$, paying attention to keep the sum $|\vec{u_f}|+|\vec{v_f}| = \text{const}$ and also $L_{tot} = \text{const}$. The figure shows that the curve $\min(\mathbf{U}_a)$ is almost a straight line with slope $m \simeq 1.15$. This means that the minimum of the anisotropy strain energy is achieved when the force ratio $\frac{\vec{v_f}}{\vec{u_f}}$ and the geometric ratio $\frac{b}{a}$ are roughly the same. Therefore, we have $\min\big[\mathbf{U}_a(\frac{\vec{v_f}}{\vec{u_f}},\frac{b}{a})\big] \rightarrow \dfrac{\vec{v_f}}{\vec{u_f}} \propto \dfrac{b}{a}$.

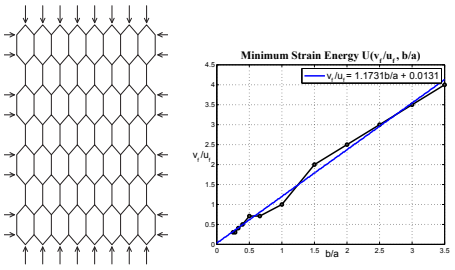
In Section 6.2 we will introduce tuning parameters that allow the user to set the range of densities and anisotropies spanned by a mesh, thus in the following we will assume that the slope m is absorbed by the anisotropy parameter and we will simply set the anisotropies of Ψ_e and Ψ_f to be equal, i.e.:

$$a_e = a_f = \frac{|u_f|}{|v_f|}. \tag{11}$$

4. Statics-aware remeshing

We exploit the non-Euclidean metric g_Ψ defined in the previous section to compute an anisotropic, density varying remeshing of surface $\mathcal{S}$. In fact, we compute a mesh that is isotropic and as regular as possible with respect to metric g_Ψ: in Euclidean space, cells of such mesh will be scaled, stretched and aligned according to the guiding tensor field Ψ_e, thus forming a grid-shell whose elements are locally close to the ideal shape as described in Section 3.2.

In order to support different meshing techniques in the same framework, we follow the approach described in [PPTSH14], transforming metric g_Ψ into a Euclidean metric on a deformed surface $\mathcal{S}'$. This metric becomes locally Euclidean if the underlying space in the neighborhood of each point p is deformed by tensor $\mathbf{W}$ computed at p, as defined in Equation 3. We evaluate $\mathbf{W}$ at each triangle of the input



(a) Anisotropic hexagonal grid subject to an *anisotropic* state of stress.

(b) Plot of the function $\min \mathbf{U}(\frac{\vec{v_f}}{\vec{u_f}}, \frac{b}{a})$.

Figure 5: Derivation of the relationship $\min\left[\mathbf{U}_a(\frac{\vec{v_f}}{\vec{u_f}}, \frac{b}{a})\right]$ for an *anisotropic* hex-grid subject to an anisotropic state of stress.

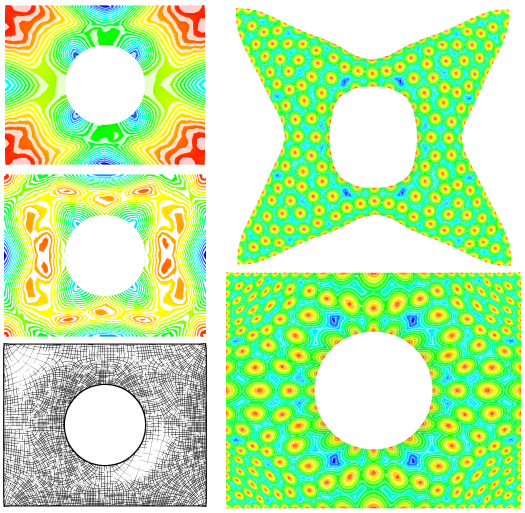


Figure 6: Density, anisotropy and directional field of the British dataset (left); the resulting deformed domain mesh (top right) and the corresponding undeformed domain (bottom right) with the resulting anisotropic, density-varying distance field for a set of samples on the surface.

mesh $\mathcal{S}$, and we resolve an optimization problem that tends to deform each triangle t to its ideal shape to make the metric Euclidean over t. See [PPTSH14] for further details, and Figure 6 for an example. Note that although this technique could have limitations under drastic deformations, our architectural surfaces always require mild deformations (see also Section 6 about bounding the dynamic range of density and anisotropy).

Once the deformed surface $\mathcal{S}'$ has been obtained, we can apply an isotropic remeshing method over it. The mesh computed in this way is finally undeformed, thus obtaining the desired anisotropic remeshing of $\mathcal{S}$.

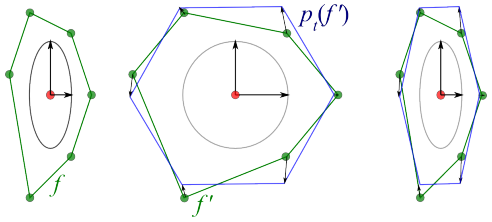


Figure 7: A single face f of the ACVT with the eigenvectors resulting form PCA (left); the un-stretched polygon f' with the aligned target polygon $p_t(f')$ (middle); and the computed displacement vectors in the original space (right).

As illustrated in Figure 12, we have experimented our framework with several remeshing methods: uniform quadrilateral meshes, computed by using [BZK09]; uniform hex-dominant meshes, obtained by using a variant of [LLW14]; and Centroidal Voronoi Tassellations, obtained by using a variant of [VC04]. Parametrisation-based methods [BZK09, LLW14] require a guiding field to be defined over the input surface. We have experimented both with the line field induced on $\mathcal{S}'$ by Ψ_e, and with a smooth cross field roughly alignd to curvature, as proposed in [BZK09]. Results obtained with the different methods are illustrated in Figure 12 and discussed in Section 6.

4.1. Regularization

In order to improve the aesthetics, as well as the planarity of faces, we optimize their shape to make them as similar as possible to stretched regular polygons. To this aim, we adopt a framework similar to [BDS*12], where we alternate *per-polygon* and *per-vertex* fitting steps.

In the per-polygon step, for each face f, we first perform a Principal Component Analysis to evaluate how much f is stretched with respect to a regular polygon. Then we compute a new polygonal region f' corresponding to f deformed (i.e., un-stretched) according to the two highest rank eigenvectors of the PCA. Next, we define a *target* regular polygon $p_t(f')$ having the same number of edges and equal perimeter as f'; then, using [BM92], we rigidly align $p_t(f')$ with f'; finally, we stretch the oriented polygon $p_t(f')$ back through the reverse deformation that was applied to f, and we use the vertices of this stretched regular polygon as target positions to displace the vertices of f. Figure 7 shows the steps of this process for a single face.

In the per-vertex fitting step, for each vertex v independently, we evaluate the position minimizing the sum of squared distances from all the target positions specified for v by its incident faces. We use a damping factor for improving convergence of this procedure.

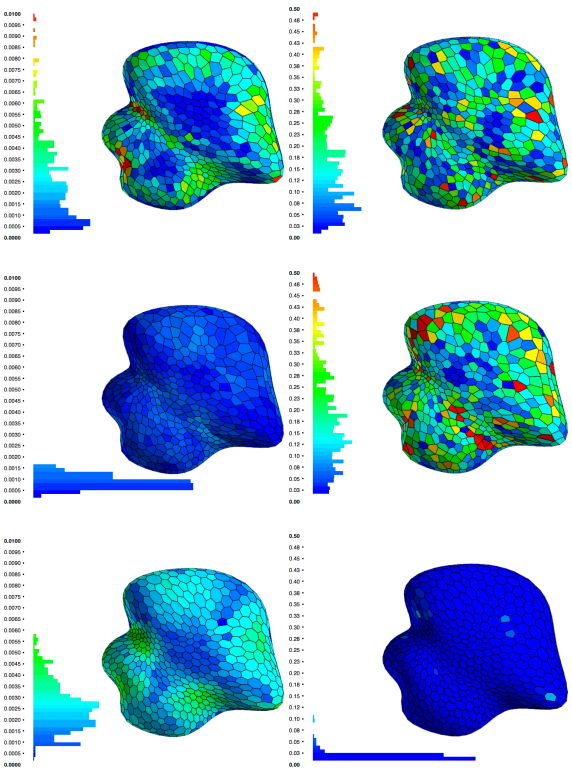


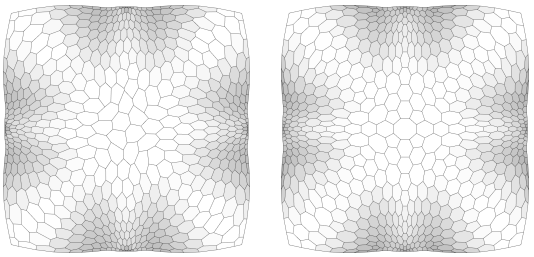
Figure 8: The effects of the regularization process on planarity (left) and regularity (right). The initial ACVT (top), optimized for planarity with Shape-Up (middle) and regularized using our procedure (bottom).

An interesting side effect of this regularization procedure is that it tends to make the length of edges of a single polygon more uniform, so that the areas of faces will vary according to the number of sides of the polygons. From an aesthetic point of view, this situation matches the look of the Archimedean class of semi-regular polyhedra.

Given the similarity of this optimization approach with [BDS*12], we have also compared our results with the planarization approach presented in that paper. Figure 8 shows our approach in comparison with our initial tessellation and with the result of Shape-Up planarization. As expected, Shape-Up achieves better planar faces, while our algorithm achieves a much better regularity of faces, hence better aesthetics. Planarity is usually measured on quad meshes as distance between diagonals divided by average edge [TSG*14]. We generalize this measure to polygonal meshes as the average distance of vertices to the best fitting plane divided by half perimeter. Regularity of a face is measured as the sum of squared distance of its vertices to their target positions, divided by its area.

Figure 8 clearly shows that our regularization algorithm yields faces that are not planar. Nevertheless, according to our definition of planarity, the average non planarity ratio for the panels of the Botanic dataset is just $\varepsilon = 0.0025$. With reference to a hypothetical isotropic hexagonal face with an edge length $L = r_{circ} = 1.5$ m, this value yields a very small average distance between the face vertices and their interpolating plane: $d_{av} = 3 \cdot L \cdot \varepsilon = 3 \cdot 1.5\ m \cdot 0.0025 = 0.01125\ m = 11.25\ mm$. Assuming, for the sake of simplicity, that we yeld single curvature for the panel, this out-of-plane measure corresponds to a radius of curvature $R \simeq 100\ m$, a huge and reassuring value if compared to the limit value $R \simeq 3\ m$ achievable with cold bent glass [BIVIC07, p.1].

Regularization also slightly improves the overall structural properties of the grid-shell structure (10% on average in our experiments). The more uniform length of edges resulting from regularization is also an advantage during production. Figure 9 shows a comparison between a meshing obtained using CVT only and the one obtained after optimisation and symmetrisation.



Non Optimized
$\lambda = 2.56\ \delta = 149.1$

Optimized
$\lambda = 2.79\ \delta = 138$

Figure 9: Comparison of non-optimized versus symmetrized and optimized tessellation of the Shell dataset. See Section 6 for the meaning of statics parameters λ and δ.

5. Implementation and user control

In most cases, the result of static analysis is not directly usable. The stress field, as obtained from FEA, may contain spikes of intensity and abrupt changes of direction of the line field. Moreover, it does not follow closely symmetries of the surface, and it may contain drastic variations of density and anisotropy that are not compatible with production. Therefore, before using field Ψ_f to generate our guiding field Ψ_e, we smooth and symmetrize it. We also allow the user to adjust the dynamic range of density and anisotropy of Ψ_e prior to generate the induced metric.

5.1. Smoothing

We extrapolate line field $\vec{u}_n$ from Ψ_f and we smooth it following [BZK09], modified as in [PLPZ12]. In short, we trade-off smoothness and faithfulness to the original line

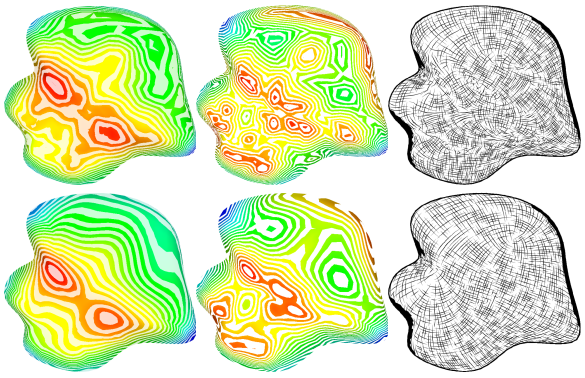


Figure 10: Smoothing and saturating the *density* (left), the *anisotropy* (middle), and the two orthogonal line fields (right) of the Botanic model: upper side original, lower side smoothed.

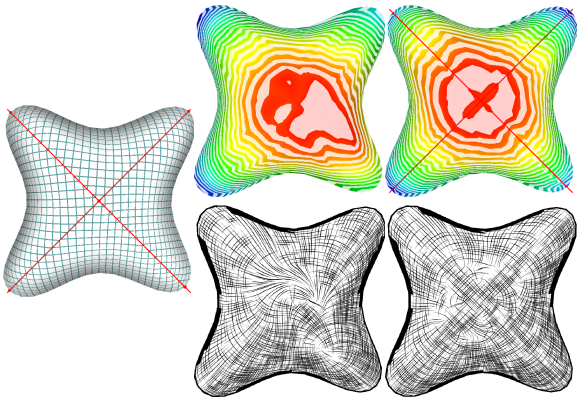


Figure 11: Symmetrization of Ψ for the Lilium dataset: on the left the cross parametrization defined by two symmetry planes; on the center/right the density field (top) and the line field (bottom), before/after symmetrization.

field, weighting the second term with anisotropy: we preserve those portions of field where there is a significant difference between the magnitudes of the two stress vectors, while obtaining a smoother field elsewhere.

We smooth the density d_f and anisotropy a_f of Ψ_f independently, by treating them as scalar fields and enforcing each of them to satisfy Lipschitz condition [BO05], i.e., $|d_f(p) - d_f(p+\vec{\varepsilon})| < L|\vec{\varepsilon}|$, with L approximately equal to the diameter of the smallest face we expect to obtain. Smoothing here is performed through an upper saturation process that preserves the extrema of the function. The results of smoothing are depicted on the lower side of Figure 10.

5.2. Symmetrization

Many architectural models present symmetry planes that should be preserved in the generated grid-shell. Assume we have one or more symmetry planes (shown in red in Fig.11) that partition the mesh into regions. We cross parameterize each symmetry region so that $\mu_{i,j}(p)$ be a cross-parametrization that maps a point p of region i onto its symmetric mate in region j. Cross-parametrizations are computed between adjacent regions in pairs and propagated about the center of symmetry. For two adjacent regions i and j, we first cross-map corresponding points on their boundaries, exploiting the common boundary along the symmetry plane, plus symmetric corners that appear along intersections with other planes of symmetry and/or sharp corners on the boundary of the object. Then we compute a harmonic map for each region onto the same parametric domain, in such a way that symmetric points are mapped to the same point in parameter space. Finally, we compute a symmetric field $\bar{\Psi}_f$ by averaging it component-wise at all the corresponding points in the various regions (see Figure 11).

5.3. Tuning anisotropy and density

Since the variation of density and anisotropy induced from Ψ_e may be not compatible with constructability of the grid-shell, we let the user adjust the desired variation of density and the desired amount of anisotropy over the surface, by introducing two parameters $D, A \geq 1$ and rescaling the density d and anisotropy a of Ψ_e in the intervals $[1, D]$ and $[1, A]$, respectively, prior to computing deformation. In Section 6, we show how such parameters can be used to fine tune the statics as well as the aesthetics of the grid-shell.

6. Implementation Details

Our method has been implemented in C++. Static analysis has been performed by using the *GSA* Finite Element Analysis software [Oas14], both on the input surface to obtain the stress tensor, and on the various grid-shells to test their structural behaviour. In particular, linear static analyses (we refer with *LS*) have been performed on the input surface (modelled as a continuous shell represented with linear triangular plate elements) in order to obtain the stress tensor. Additionally, geometrically non-linear elastic analyses (we refer with *GNA*) have been carried out on the grid-shells (where each edge was modelled as a beam element) in order to get the equilibrium path and the representative parameters (λ, δ) at collapse. These measures, which are most relevant in structural engineering, are two scalars defined as follows [Cri91]:

- The non-linear buckling multiplier λ, which measures the ability of a structure to bear the applied load $\mathbf{q}$ (equal to a unitary uniformly distributed load in this case) before collapsing. It measures the robustness of the structure and therefore it should be ideally maximized;
- The nodal displacement δ, which measures the maximum nodal distance between the deformed shape and the ref-

erence shape when the structure is standing under the applied load $\lambda\mathbf{q}$. It measures the compliance of the structure and it should be ideally minimized.

For the sake of completeness, a brief description of LS and GNA analysis within the Finite Element Method context is provided in Appendix A.

6.1. Meshing technique

In order to compare the different meshing techniques described in Section 4, we also measure how much each remeshing deviates from the target metric g_Ψ. To this aim, we define an error measure that exploits a variation of deformation process of regularization (see Section 4.1). We deform each polygon of the output mesh by using the inverse metric g_Ψ^{-1} (instead of using the eigenvectors of the PCA) and we take the residual error due to the rigid alignment of such deformed polygon to the target regular polygon (see Figure 7). Note that this measure takes into account both density and anisotropy.

In order to respect the scale and anisotropy induced by Ψ_e in physical space, we need a meshing that is as uniform as possible in deformed space. It is well known that this is possible only if singularities are placed at points with high Gaussian curvature, otherwise big distortions appear in parametrization. By following curvature in deformed space, instead of the direction expressed by Ψ_e directly, we trade alignment for better consistency of density and anisotropy. Notice that alignment to a line field would generate singularities with index $1/2$ and index $-1/2$ only; however, such singularities would correspond to vertices with valence 2 and 6 during meshing. In quad meshing, valence $3-5$ irregular vertices are highly preferred over vertices of valence $2-6$. For this reason, we mildly smooth the field in deformed space by treating it as a cross field, while using the input field as a soft constraint: such smoothing tends to separate index $1/2$ and $-1/2$ singularities into pairs of index $1/4$ and $-1/4$ singularities, respectively.

Instead, to generate regular hexagonal meshing, we used the method of [LLW14] which computes a quad meshing based on a line field. Again, we compute a line field which is aligned to the curvature direction on the deformed space, then we extract the hexagonal meshing.

The bottom part of Figure 12 shows the residual error for each remeshing technique on the Neumünster dataset. From this picture we may observe that the parametrisation-based methods that align the elements to the curvature of the deformed shape $\mathcal{S}'$ - namely, Figures 12(b) and 12(d) - adapt to the target metric better than the ones that align directly to the projection of Ψ_e on $\mathcal{S}'$ - namely, Figures 12(a) and 12(c). This result, which seems contradictory, comes from distortion introduced by parametrization-based remeshing methods. Only an isometric parametrization allows extracting a uniform tessellation of $\mathcal{S}'$, hence a tessellation of $\mathcal{S}$ that respects the target metric, while distortions in the parametrization result into deviations from the metric. Parametrization is much closer to isometry if the guiding field follows the geometry of $\mathcal{S}'$ (namely, its curvature) rather than the (arbitrary) field Ψ_e projected on it. We experimented this behavior consistently with all datasets, thus in the following we will just consider parametrization-based methods based on curvature in deformed space. Remeshing based on CVT, on the contrary, does not suffer from these drawbacks, since it is performed directly in physical space $\mathcal{S}'$ and it is isotropic in nature. Figure 12(e) shows that it matches rather well the desired metric.

No single method between those corresponding to cases (b), (d) and (e) of Figure 12 gives the best results for all datasets, therefore we have experimented with all three methods. To support the validity of our approach, we observe that in all experiments the statics parameters (buckling multiplier and nodal displacement) become consistently better as the meshing better adapts to the target metric.

6.2. Tuning parameters

As we mentioned in section 5.3, our method works on two parameters that must be set by the user: the threshold for density D and the threshold for anisotropy A. Comparative tests of static analysis require that different structures have the same weight and total length of beams [MW13]. We tolerate a 5% of total length variation.

We test how the variation of density and anisotropy influence the buckling multiplier λ and nodal displacement δ. We show experiments on a funicular surface: the Shell dataset. We vary density and anisotropy within a range that goes from 1 to 4, with unit step, for a total of 16 test cases. For this experiment we have used CVT meshing.

Some pictures illustrating the experiment are shown in Figure 13 (top views and graphs). Note that all our solutions outperform the isotropic meshing obtained with $D = A = 1$. However, a too large value of D and A may result in a deterioration of the static performance. In the examples shown in the next sections we keep the parameters in a reasonable range, showing that we always improve significantly the static performances with respect to state-of-the-art remeshing methods.

7. Results

We compared our grid-shells with some quadrilateral meshes obtained with [TSG*14, VHWP12]. As for the previous experiments, we set our parameters to match the total length of edges of the structures we compare with. The experiments are summarized in Table 2 and the related meshes are shown in Figure 16.

We present two different tessellations for each surface:

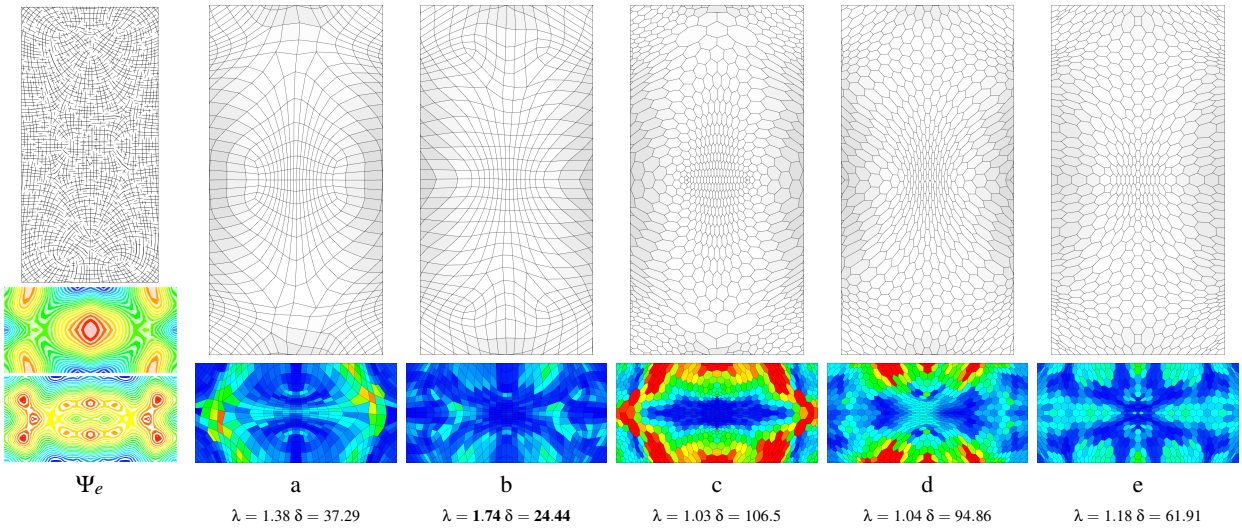


Figure 12: On the left, the guiding field (line field, density and anisotropy) for the Neumunester dataset. On the right, results different meshing techniques used in our framework: Quad meshing using [BZK09] following Ψ_e (a); Quad meshing using [BZK09] following curvatures on deformed space (b); Hexagonal meshing using [LLW14] following Ψ_e (c); Hexagonal meshing using [LLW14] following curvature directions (d); Hex dominant CVT tessellation using [VC04]. The meshing have approximately the same total edge length. The colored diagrams depict closeness of mesh faces to prescribed density and anisotropy on a blue-to-red error scale. Parameters of buckling and deformation are given below.

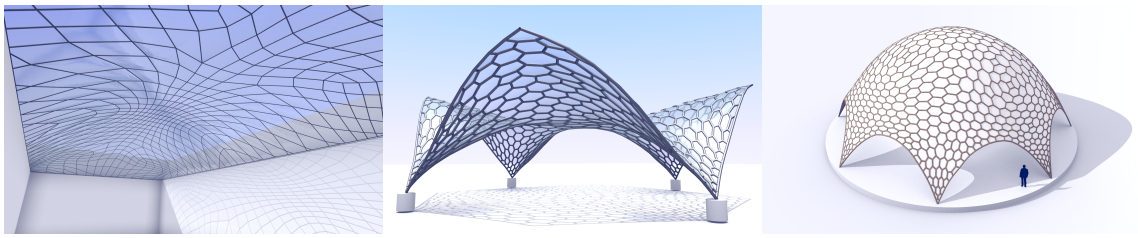
Dataset	Model	# Vertices	# Faces	# Edges	Total length (m)	λ	δ
Aquadom	[VHWP12]	1078	1004	2074	3906	1.38	164.1
	Hex (3,3)	2374	1161	3536	4036	2.72	117.1
	Quad (3,2)	1223	1133	2357	3994	2.34	129.9
Botanic	[TSG*14]	1121	1076	2196	1989	0.83	335.4
	Voronoi (3,3)	2352	1177	3528	2016	1.48	171.6
	Quad (4,4)	1071	1005	2075	1994	1.65	198.7
British	[TSG*14]	1648	1568	3216	4286	2.70	27.9
	Hex (2,2)	3219	1588	4807	4079	3.0	19.44
	Voronoi (3,3)	3460	1728	5188	4114	3.23	26.7
Lilium	[VHWP12]	1648	636	3216	1147	1.42	77.6
	AnisoQuad (3,3)	650	592	1241	1229	2.75	46.48
	Voronoi (3,3)	1444	723	2166	1182	2.97	37.55
Paraboloid	Voronoi (3,3)	1904	949	2852	2691	2.3	74.55
Shell	Voronoi (3,3)	1192	593	1784	480	2.79	138.0

Table 2: Statistics on datasets and results: for each dataset we show statistics on models taken for comparison and models built with the proposed approach. Models from [TSG*14, VHWP12] are quad meshes. Quad and Voronoi refer to our models of anisotropic quad meshes and ACVT, respectively, computed with parameters (D,A). For each model we report: the number of vertices, faces and edges; the total length of beams in the model; the buckling factor λ; and the nodal displacement δ.

meshing techniques have been selected by considering both faithfulness to the target metric and aesthetic criteria.

The setup of the each experiment is pretty realistic. We tuned the parameters by considering the size of each model. The setup is summarized in table 1. Our statics aware grid-shells always outperform state-of-the-art models, in terms of both buckling and displacement.

Figure 15 shows the effect of tessellation on the structural behavior of the grid-shell. In the Lilium dataset, the forces flow from the top to the restraints along the red paths of structural elements; in our models, such paths are better distributed than the ones in [VHWP12], thus reducing the elastic strain energy, as well as the maximal displacement. In the British Quad dataset, almost all the beams of our



$\lambda = 1.74\ \delta = 24.44$ $\lambda = 2.79\ \delta = 138.00$ $\lambda = 2.3\ \delta = 74.55$

Figure 14: Renderings of some of the models produced. From the left: Neumünster, Shell, Paraboloid.

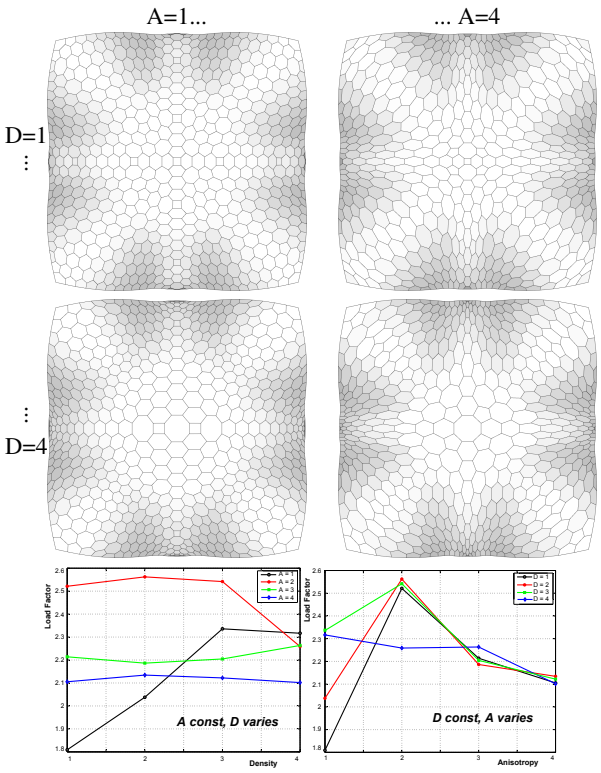


Figure 13: 4×4 test on the Shell model.

model undergo the same axial force, whereas in the model of [TSG*14] there is a strong variance of axial forces, including strong compressions (red).

7.1. Load tests on a physical replica

We have fabricated a reduced scale model of the Shell structure composed of 465 joints, 697 beams and 462 panels. The side of this reproduction is 2.4 meters. Each joint has been produced independently using a FDM printer; sticks of wood simulate beams; external panels are made of PET (Polyethylene terephthalate) and they have been laser-cut. Each component of the structure has a physical label (3D printed on joints, carved by laser on panels or glued paper on sticks), mentioned in an assembly map to support the manual assembly phase. The panels have been fixed by screwing a flat washer at each joint. Some particulars on the replica are shown in Figure 17.a.

Surface	Beams	q_{dead}	q_{glass}	q_{snow}
Shell	ϕ (110, 25)	0.90	0.75	1.00
Neumünster	ϕ (60,10)	0.50	0.75	1.00
British	ϕ (100,15)	0.65	0.75	1.00
British tri	ϕ (130, 20)	1.00	0.75	1.00
Paraboloid	ϕ (130, 20)	0.75	0.75	1.00
Aquadom	ϕ (100,15)	0.75	0.75	1.00
Botanic	ϕ (70, 15)	0.65	0.75	1.00
Lilium	ϕ (50, 10)	0.35	0.75	1.00

Table 1: Static analyses setup: For each model, the second column reports the diameter and the thickness of the steel beams in mm. The third, forth and fifth columns report the load induced by each component of the simulated scenario in (kN/m^2): beams, glass and uniformly distributed load on the surface (e.g. snow).

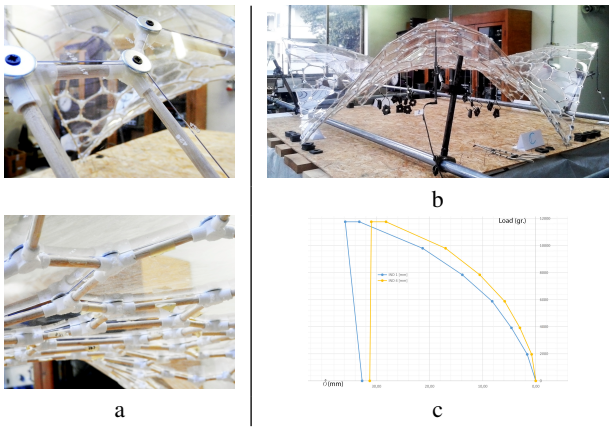


Figure 17: a. Some particular of the physical replica ; b. Setup for load tests over the fabricated model; c. displacement / external load plot.

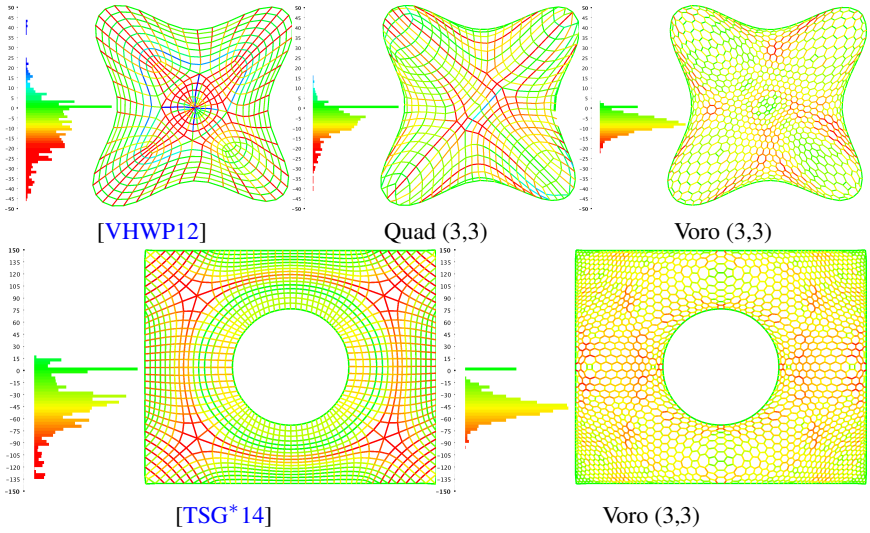


Figure 15: Comparisons of distribution of AxialForces on the datasets. Red corresponds to compression, blue corresponds to traction.

We have performed load tests on the physical structure, by incrementally applying weights and monitoring the displacement of the corners of the structure with a proper sensor (see Figure 17.b). The result of this experiment is shown in the graphs of Figure 17.c.

As shown in Figure 18, there is an excellent agreement between the predicted and the experienced results. In fact, the global deformed shape which occurs in correspondence of the buckling of the structure, is nearly the same on both the numerical and the real mode.

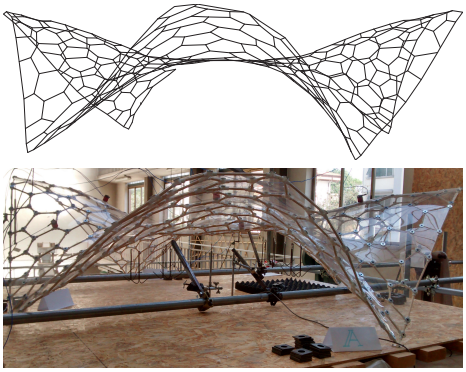


Figure 18: Buckling shape of the mockup: optimal agreement between numerical prediction (Top) and experimental results (Bottom).

8. Concluding remarks

We have presented a practical framework that, given a fixed shell shape, allows us to generate grid-shell structures with static performances that are better than the current state of the art.

We start from a FEM stress analysis of the input surface and we define a method to convert the computed stress field into an anisotropic, density-varying, non-Euclidean metric that we use to drive the remeshing of the input surface. We derive this new anisotropic metric so that the shape and the size of the generated cell is locally optimized for stiffness w.r.t. to the computed stress. We have applied this framework to various tessellation methods to generate hex-dominant as well as quad meshes optimized for good static behavior.

We have validated our claims by means of simulation as well as experimentally. We have tested the generated structures evaluating, by means of industrial standard non-linear analysis simulations, their behavior in terms of non-linear buckling multiplier and nodal displacement. Moreover we have built a reduced scale model and we have performed physical tests on it to verify the soundness of the behavior predicted by the simulation. The result of our experiments demonstrate that our grid-shells achieve significantly better static performances with respect to state-of-the art grid-shells.

Our method is fairly general and it can be applied to generic architectural surfaces, including non-funicular surfaces. Results can be deteriorated in the presence of drastic deformations due to too high dynamics in either density or anisotropy. This is not realistic scenario, though, because drastic variations in size and/or highly elongated shapes do not lend themselves to practical construction. For this reason, we have inserted user parameters to set the desired level of density and anisotropy.

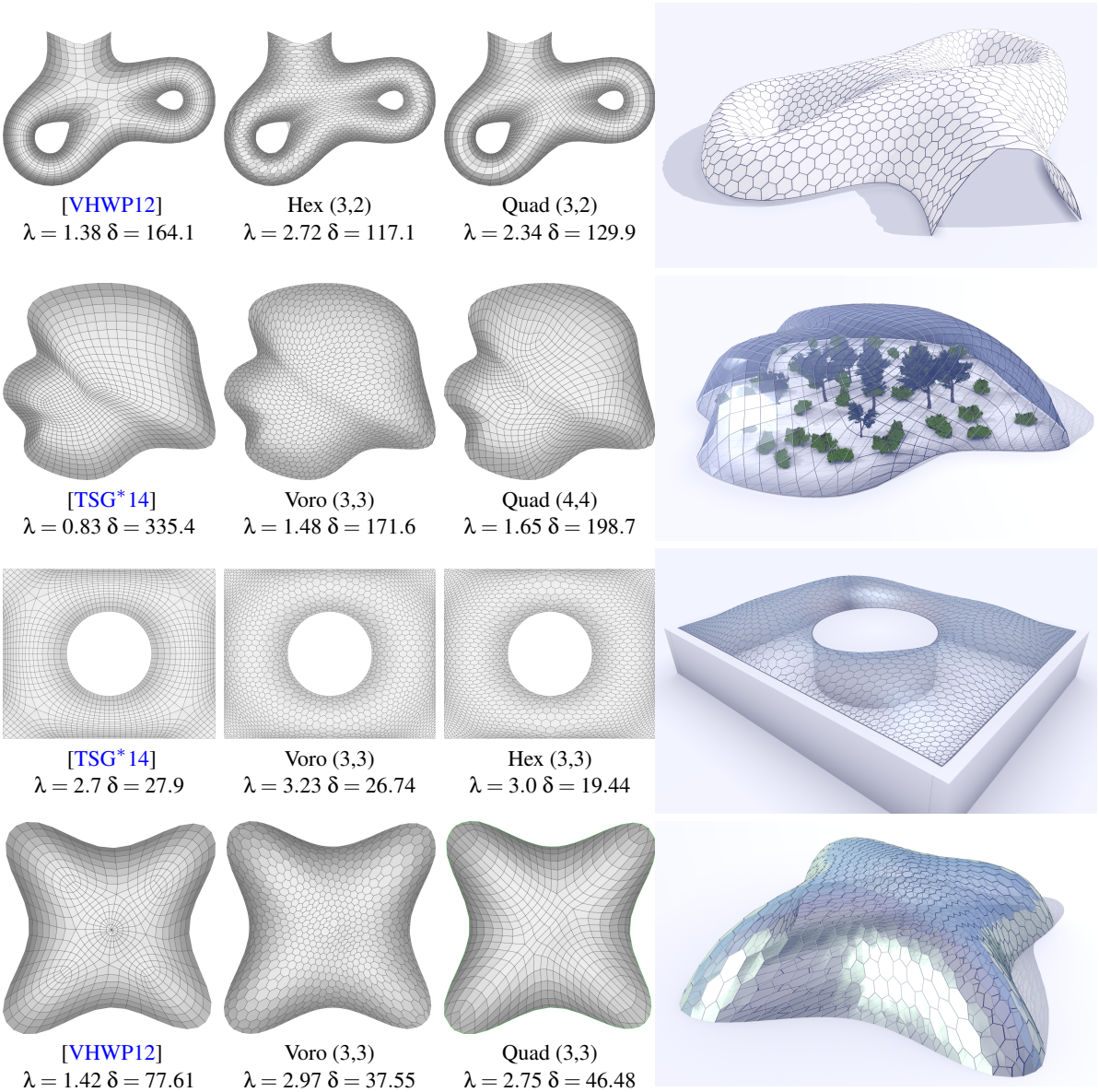


Figure 16: Comparison with state-of-the art remeshings, top views. From the top: Aquadom, Botanic, British, Lilium.

Our method does not guarantee optimality. Further improvement could be obtained by geometric optimization that moves the vertices of the mesh tangentially, without changing either the surface or the topology, while seeking for a better distribution of forces. We plan to tackle this problem in future work.

Other interesting extensions concern rationalization, by exploiting the tendency of our shape optimization algorithm to produce cells close to Archimedean solids. Moreover we can also extend our regularization technique to produce single-curvature panels that can be produced with cold bent glass.

Acknowledgements

The research leading to these results was funded by EU FP7 project ICT FET Harvest4D (http://www.harvest4d.org/, G.A. no. 323567). We would like to thank Marco Tarini for his support on hex mesh processing, Carmelo Di Maria for his support on the physical replica, Daniele Panozzo for

providing the code of Libigl [JP*13], Etienne Vouga and Chengcheng Tang for providing data for comparisons.

Appendix A: Details on Finite Element Analysis

The *displacement method* is used to numerically solve problems in elasticity. The structure is represented as an assembly of parts (i.e. finite elements, trusses, beams, plates, shells...), whose stress/strain state is determined by their nodal displacements together with their shape functions [Bat96, p. 149]. The global equilibrium equations are formulated in terms of the displacements $\mathbf{p}$ of the nodal coordinates of the system, its global stiffness matrix $\mathbf{K}$ and the applied forces $\mathbf{q}$ [Bat96, p. 152]:

$$\mathbf{K}\mathbf{p} = \mathbf{q}. \tag{12}$$

The stiffness matrix $\mathbf{K}$ depends only on geometry of the mesh and on material properties. The nodal displacements represent the solution of the problem: they are stored in a $6 \times n$ matrix $\mathbf{p}$, where n is the total number of nodes composing the structural model and each node has three translational and three rotational degrees of freedom. Once displacements are known, the stress tensor can be easily computed by means of the elements shape functions and the generalized Hooke's Law: a first order linear approximation of the relationship between strains (i.e. deformations as the first order derivatives of nodal displacements $\mathbf{p}$) and stresses. Stresses σ_{ij}, strains ε_{kl} and Hooke's law c_{ijkl} are two second order tensors and one fourth order tensor, respectively: $\sigma_{ij} = c_{ijkl}\varepsilon_{kl}$, where the Einstein summation convention is intended.

When the system is linear or the applied load is small, a Linear Static analysis (LS) is sufficiently accurate and it consists in just applying equation 12. It yields the displacements of the nodal coordinates $\mathbf{p}$ as well as the stress tensor for each element. Instead, when the system displays high non-linearities (that is the case of grid-shells loaded up to collapse), LS analysis is no more reliable and Geometrically Non-Linear Analysis (GNA) is adopted. This well-known analysis consists in repeatedly applying equation 12 within a Newton-Raphson iteration scheme:

$$\mathbf{K}_r\left(\mathbf{p}_{r+1} - \mathbf{p}_r\right) = \mathbf{q}_i - \mathbf{q}_r \tag{13}$$

where $\mathbf{K}_r$ is the tangent stiffness matrix at the r-*th* iteration, $\mathbf{p}_{r+1}$ and $\mathbf{p}_r$ are the displacement vectors at the (r+1)-*th* and r-*th* iterations, respectively, $\mathbf{q}_i$ is the external load vector in correspondence of the i-*th* load increment and $\mathbf{q}_r$ is the internal forces vector at the r-*th* iteration. Stiffness matrix $\mathbf{K}_r$ and loads $\mathbf{q}_r$ are recomputed by updating mesh geometry with displacements $\mathbf{p}_r$. Equation 13 can be rewritten in explicit form as:

$$\mathbf{p}_{r+1} = \mathbf{p}_r + \mathbf{K}_r^{-1}\left(\mathbf{q}_i - \mathbf{q}_r\right) \tag{14}$$

that is solved numerically until the norm $||\mathbf{q}_i - \mathbf{q}_r||$ is within acceptable limits. At each step the load q_i is incremented of a certain amount $\Delta\mathbf{q}_i$ (manually or automatically determined) and equation 14 is applied. If the aforesaid norm is low enough, then another load increment $\Delta\mathbf{q}_{i+1}$ can be applied, otherwise the current increment gets halved and equation 14 is applied again until the convergence condition is met. In particular, the structure is deemed to be collapsed when no convergence can be achieved in a certain number of iterations within the same load increment. Hence GNA requires several runs of LS analysis, in each of which the tangent stiffness matrix must be built (as the geometry of the structure changes at each load step) and inverted, therefore it is an highly time consuming task. Nevertheless, with this technique the non-linear behavior of the structure under loading can be closely followed up to collapse.

In summary:

- The non-linear buckling multiplier λ is the summation of all the load increments $\Delta\mathbf{q}_i$ applied in a GNA analysis up to collapse;
- The displacement δ mentioned in Section 6 is a user selected single entry of the $6 \times n$ displacements matrix $\mathbf{p}$ at time of collapse. Usually this is chosen as the z displacement of a specific node which is highly representative of the structure's behavior (e.g. the pole of a dome, the key brick of an arch);
- The relationship between δ and λ is non-linear and is numerically described by one of the $6 \times n$ scalar equations expressed by the vector equation 14, in particular that referring to the nodal coordinate associated to δ.

For further details see [Bat96, Cri91, Cri97].

References

[Bat96] BATHE K.-J.: *Finite Element Procedures*. Englewood Cliffs, N.J. Prentice Hall, 1996. 3, 14

[BDS*12] BOUAZIZ S., DEUSS M., SCHWARTZBURG Y., THIBAUT W., PAULY M.: Shape-Up: Shaping Discrete Geometry with Projections. *Computer Graphics Forum (proc of SGP 2012) 31*, 5 (2012). 6, 7

[BIVIC07] BELIS J., INGHELBRECHT B., VAN IMPE R., CALLEWAERT D.: Experimental assessment of cold-bent glass panels. *Glass Performance Days, Tamglass 1* (2007), 115–117. 7

[BK01] BULENDA T., KNIPPERS J.: Stability of grid shells. *Computers and Structures 79* (2001). 2

[Blo09] BLOCK P.: *Thrust network analysis: exploring three-dimensional equilibrium*. PhD thesis, Massachusetts Institute of Technology, 2009. 2

[BM92] BESL P. J., MCKAY N. D.: A method for registration of 3-D shapes. *IEEE transactions on pattern analysis and machine intelligence 14*, 2 (1992), 239–256. 6

[BO05] BOISSONNAT J.-D., OUDOT S.: Provably good sampling and meshing of surfaces. *Graph. Models 67*, 5 (Sept. 2005), 405–451. 8

[BZK09] BOMMES D., ZIMMER H., KOBBELT L.: Mixed-integer quadrangulation. *ACM Transactions on Graphics 28*, 3 (July 2009), 1. 6, 7, 10

[Cri91] CRISFIELD M.: *Non-linear finite element analysis of solids and structures: Essentials*. John Wiley & Sons, Inc., Chichester, New York, 1991. 8, 14

[Cri97] CRISFIELD M. A.: *Non-Linear Finite Element Analysis of Solids and Structures: Advanced Topics*, 1st ed. John Wiley & Sons, Inc., New York, NY, USA, 1997. 14

[CW07] CUTLER B., WHITING E.: Constrained planar remeshing for architecture. In *Graphics Interface* (2007), pp. 11–18. 2, 3

[EKS*10] EIGENSATZ M., KILIAN M., SCHIFTNER A., MITRA N. J., POTTMANN H., PAULY M.: Paneling architectural freeform surfaces. *ACM Trans. Graph. 29*, 4 (2010), 45:1–45:10. 2

[FLHCO10] FU C.-W., LAI C.-F., HE Y., COHEN-OR D.: K-set tilable surfaces. *ACM Trans. Graph. 29*, 4 (2010), 44:1–44:6. 2

[JP*13] JACOBSON A., PANOZZO D., ET AL.: libigl: A simple C++ geometry processing library, 2013. http://igl.ethz.ch/projects/libigl/. 14

[JWWP14] JIANG C., WANG J., WALLNER J., POTTMANN H.: Freeform honeycomb structures. *Computer Graphics Forum 33*, 5 (2014). Proc. Symp. Geom. Processing. 1, 2

[LLW14] LI Y., LIU Y., WANG W.: Planar hexagonal meshing for architecture. *IEEE Transactions on VIsualization and Computer Graphics* (2014). In press. 6, 9, 10

[LPS*13] LIU Y., PAN H., SNYDER J., WANG W., GUO B.: Computing self-supporting surfaces by regular triangulation. *ACM Trans. Graph. 32*, 4 (July 2013), 92:1–92:10. 2

[LPW*06] LIU Y., POTTMANN H., WALLNER J., YANG Y.-L., WANG W.: Geometric modeling with conical meshes and developable surfaces. *ACM Trans. Graph. 25*, 3 (2006), 681–689. 2

[LSZ*14] LU L., SHARF A., ZHAO H., WEI Y., FAN Q., CHEN X., SAVOYE Y., TU C., COHEN-OR D., CHEN B.: Build-to-last: Strength to weight 3d printed objects. *ACM Trans. Graph.* (2014). to appear. 2

[LXW*11] LIU Y., XU W., WANG J., ZHU L., GUO B., CHEN F., WANG G.: General planar quadrilateral mesh design using conjugate direction field. *ACM Trans. Graph. 30*, 6 (2011), 140:1–140:10. 2, 3

[MW13] MALEK S., WILLIAMS C.: Structural implications of using cairo tiling and hexagons in gridshells. In *Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium 2013* (2013). 2, 9

[Oas14] OASYS: Gsa analysis, 2014. http://http://www.oasys-software.com. 8

[OKF08] OGAWA T., KATO S., FUJIMOTO M.: Buckling load of elliptic and hyperbolic paraboloidal steel single-layer reticulated shells of rectangular plan. *Journal of the International Association for Shell and Spatial Structures* (2008). 2

[OR95] OTTO F., RASH B.: *Finding Form*. Edition Alex Menges, Stuttgart, 1995. 1, 2

[PBSH13] PANOZZO D., BLOCK P., SORKINE-HORNUNG O.: Designing unreinforced masonry models. *ACM Trans. Graph. 32*, 4 (2013), 91:1–91:12. 2

[PJH*14] POTTMANN H., JIANG C., HÖBINGER M., WANG J., BOMPAS P., WALLNER J.: Cell packing structures. *Computer-Aided Design* (Mar. 2014), 1–14. 1, 2, 3

[PLPZ12] PANOZZO D., LIPMAN Y., PUPPO E., ZORIN D.: Fields on symmetric surfaces. *ACM Trans. Graph. 31*, 4 (2012), 111:1–111:12. 7

[PLW*07] POTTMANN H., LIU Y., WALLNER J., BOBENKO A., WANG W.: Geometry of multi-layer freeform structures for architecture. *ACM Trans. Graphics 26*, 3 (2007). 3

[PPTSH14] PANOZZO D., PUPPO E., TARINI M., SORKINE-HORNUNG O.: Frame fields: Anisotropic and non-orthogonal cross fields. *ACM Trans. Graph.* (2014). to appear. 5, 6

[SB10] SCHIFTNER A., BALZER J.: Statics-sensitive layout of planar quadrilateral meshes. In *Advances in Architectural Geometry 2010*, Ceccato C., Hesselgren L., Pauly M., Pottmann H., Wallner J., (Eds.). Springer Vienna, 2010, pp. 221–236. 2, 3

[SHWP09] SCHIFTNER A., HÖBINGER M., WALLNER J., POTTMANN H.: Packing circles and spheres on surfaces. *ACM Trans. Graph. 28*, 5 (2009), 139:1–139:8. 2

[SS10] SINGH M., SCHAEFER S.: Triangle surfaces with discrete equivalence classes. *ACM Trans. Graph. 29*, 4 (2010), 46:1–46:7. 2

[TG69] TIMOSHENKO S., GOODIER J.: *Theory of elasticity*. McGraw-Hill classic textbook reissue series. McGraw-Hill, 1969. 2

[Tro08] TROCHE C.: Planar hexagonal meshes by tangent plane intersection. *Advances in Architectural Geometry 1* (2008), 57–60. 2

[TSG*14] TANG C., SUN X., GOMES A., WALLNER J., POTTMANN H.: Form-finding with polyhedral meshes made simple. *ACM Trans. Graph.* (2014). to appear. 2, 3, 7, 9, 10, 11, 12, 13

[TWK87] TIMOSHENKO S. P., WOINOWSKI-KRIEGER S.: *Theory of plates and shells*. Mc Graw Hill classic textbook reissue. Mc Graw Hill, New York, 1987. 2

[VC04] VALETTE S., CHASSERY J.-M.: Approximated Centroidal Voronoi Diagrams for Uniform Polygonal Mesh Coarsening. *Computer Graphics Forum 23*, 3 (Sept. 2004), 381–389. 6, 10

[VHWP12] VOUGA E., HÖBINGER M., WALLNER J., POTTMANN H.: Design of self-supporting surfaces. *ACM Trans. Graph. 31*, 4 (2012), 87:1–87:11. 9, 10, 12, 13

[YYPM11] YANG Y.-L., YANG Y.-J., POTTMANN H., MITRA N. J.: Shape space exploration of constrained meshes. *ACM Trans. Graph. 30*, 6 (2011), 124:1–124:12. 2

[ZCBK12] ZIMMER H., CAMPEN M., BOMMES D., KOBBELT L.: Rationalization of triangle-based point-folding structures. *Comp. Graph. Forum 31*, 2pt3 (2012), 611–620. 2

[ZSW10] ZADRAVEC M., SCHIFTNER A., WALLNER J.: Designing quad-dominant meshes with planar faces. *Computer Graphics Forum 29*, 5 (2010), 1671–1679. Proc. Symp. Geometry Processing. 2, 3