Polyhedral Patterns

Caigui Jiang* Chengcheng Tang* Amir Vaxman Peter Wonka† Helmut Pottmann†
King Abdullah University of Science and Technology (KAUST) TU Wien / Utrecht University KAUST TU Wien / KAUST

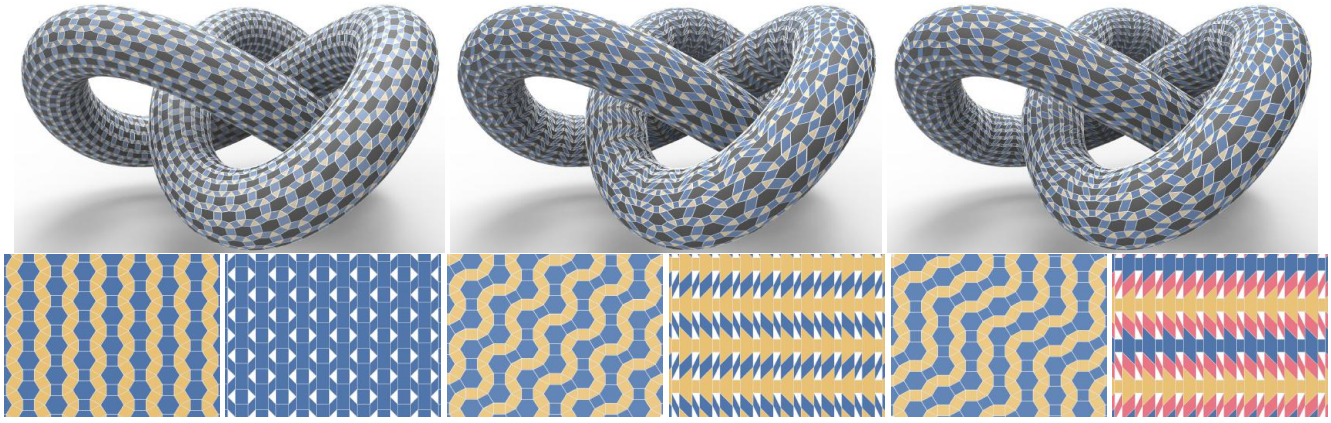


Figure 1: *Polyhedral patterns on a knot. Top: three polyhedral patterns tiling a knot and optimized by our framework. All examples are combinatorially equivalent to a semi-regular pattern* $(3, 4, 6, 4)$*. Bottom: each of the three solutions is induced by a choice of* strip decomposition *and corresponding* affine symmetries*. For each model, we show the strip decomposition (left) with the pattern in the plane colored by yellow and blue strips. We show the deformed pattern upon mapping to a cylinder, suggesting the feasible symmetries (right). The different colors encode different choices of symmetries. For instance, blue faces are symmetric with respect to the barycenter.*

Abstract

We study the design and optimization of polyhedral patterns, which are patterns of planar polygonal faces on freeform surfaces. Working with polyhedral patterns is desirable in architectural geometry and industrial design. However, the classical tiling patterns on the plane must take on various shapes in order to faithfully and feasibly approximate curved surfaces. We define and analyze the deformations these tiles must undertake to account for curvature, and discover the symmetries that remain invariant under such deformations. We propose a novel method to regularize polyhedral patterns while maintaining these symmetries into a plethora of aesthetic and feasible patterns.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

Keywords: Architectural geometry, discrete differential geometry, polyhedral meshes, wallpaper tiling groups

* Joint first authors.
† Joint last authors.

1 Introduction

Architects and engineers are constantly pushing design boundaries by exploring new building shapes and modeling their appearances. Advances in architectural geometry have made it possible for many buildings to be shaped as freeform surfaces. To conform to construction constraints, such designs are often rationalized with *meshes* that have planar faces. These faces are then realized with common materials, such as wood (see Fig. 3) or glass.

Symmetric tessellation patterns have often been used in art, architecture, and product design for their aesthetic merits. However, the use of these patterns was restricted to planar surfaces, such as windows, walls, or floors. Notable examples are Arabesques, stained-glass patterns, and mosaics [Abas et al. 1995; Lu and Steinhardt 2007]. Here, we seek to enrich architectural design by meshing freeform surfaces with tessellation patterns. Examples of those are in Figures 1 and 2.

There are several key challenges in designing polyhedral patterns on curved freeform surfaces. Simply placing a given pattern on such a surface (e.g., by a parametrization) and optimizing for tile planarity without any regularization of tile shapes is bound to fail; planarity alone is severly underconstrained, and the process is apt to degenerate to solutions with zero-length edges or foldovers (see Figure 4). Moreover, an arbitrary choice of tile-shape regularization with commonly used measures, such as face or angle distortion, Laplacian smoothness, or edge-length preservation, might clash with the planarity constraint, resulting in over-constrained optimization. As a consequence, either the mesh is not planarized, or the desired regularity is not satisfied. (Figures 27 and 28 show examples.)

Our solution to the problem is to study explicit constructions of polyhedral patterns that approximate surfaces with varying Gaussian curvature. We observe curvature-invariant regularities, namely different types of symmetries. We introduce a theoretical study of polyhedral patterns that explains our choice of regularizers, and which leads to an objective function that is neither over- nor under-constrained.

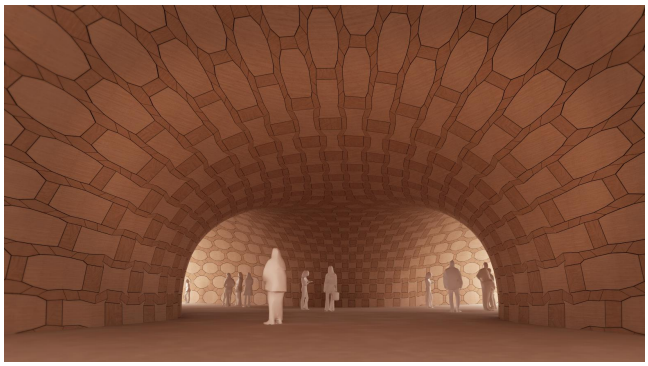


Figure 2: *Cladding an interior space with a polyhedral pattern using wooden panels. The pattern transforms smoothly from positive to negative curvature regions.*

Figure 3: *A planar-hexagonal pavilion constructed with wooden panels [Krieg et al. 2014].*

We show a set of results that demonstrate, for the first time, the computation of such patterns on surfaces that satisfy both planarity and regularity constraints. We focus on *semi-regular patterns* (see Figure 5), which are patterns comprising regular polygons.

Our contributions are:

- An analysis of surface approximation with polyhedral patterns by describing tile shape deformations, in order to accommodate for curvature. Consequently, we show how different *strip decompositions* result in a variety of patterns.
- Affine symmetries that are curvature-invariant. We construct a family of *regularizers* encoding such symmetries: e.g., symmetries with respect to axes passing through vertices, edge midpoints, or face barycenters, and reflective symmetries with respect to planes.
- A variety of polyhedral patterns that have not been demonstrated before.

2 Previous Work

Approximation with polyhedral meshes can be achieved with variational shape approximation [Cohen-Steiner et al. 2004]. However, the resulting unstructured mesh is built to satisfy required approximation accuracy and does not follow a prescribed pattern.

Most previous works that focused on polyhedral mesh creation targeted *planar quad* (PQ) meshes. PQ meshes play a central role in discrete differential geometry [Sauer 1970; Bobenko and Suris 2008], and have attracted considerable interest in recent years, cf. [Liu et al. 2006; Liu et al. 2011; Zadravec et al. 2010], as their design is a core problem in architectural geometry.

Planar hexagonal (PH) meshes have also been studied, but to a lesser extent. The simplest way to produce them is by taking the dual of

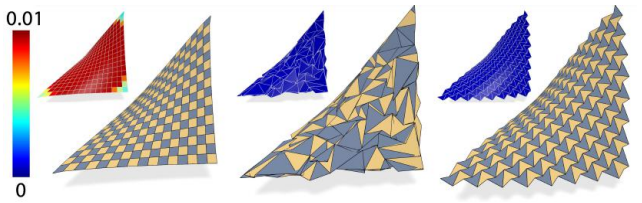


Figure 4: *Under- and over-constrained optimization illustrated on a quad mesh. The initial mesh is aligned with the parameter lines on a bilinear surface. The task is to planarize it. Left: Using mesh polyline smoothness as a regularizer results in an over-constrained problem, and acceptable planarity is not achieved (red color). Middle: Dropping the regularizer leads to an under-constrained problem, where the faces are perfectly planar, yet their appearance is chaotic and unaesthetic. Right: Using our regularizer, based on affine symmetries (with respect to edge midpoints), yields an aesthetically pleasing pattern with planar quads.*

triangle meshes [Zimmer 2014; Li et al. 2014] or remeshing triangle meshes by parametrization and deformation [Vaxman and Ben-Chen 2015]. They are problematic in several aspects: the face shapes have to change considerably and even become concave in negatively curved areas. In addition, they have to transition smoothly between regions of negative and positive Gaussian curvature. We present a systematic way to regularize patterns between different curvature regions, including PH meshes as a special case.

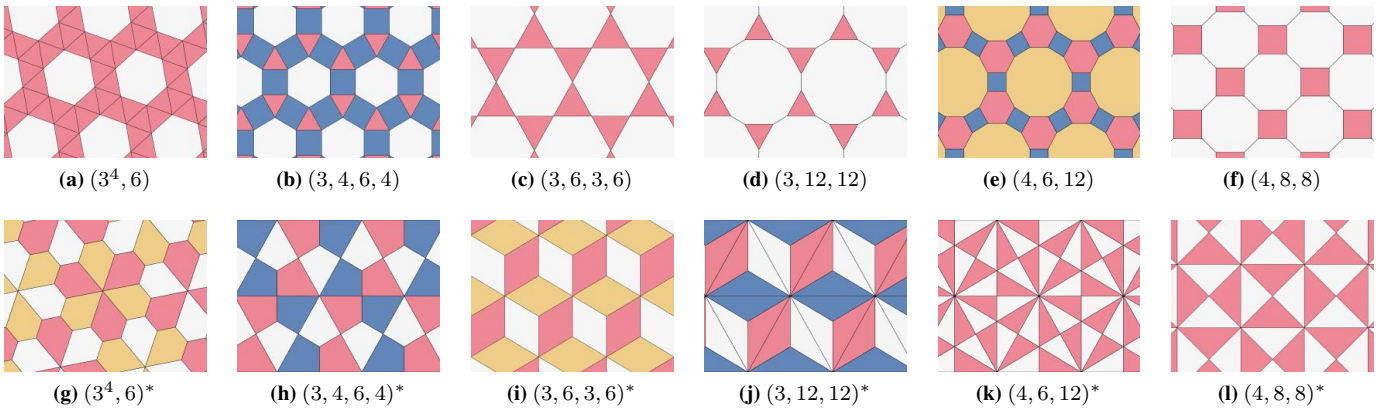
Special polyhedral patterns appear as by-products with circle-packing meshes [Schiftner et al. 2009] and special hexagonal support structures [Jiang et al. 2014]. However, these papers do not consider polyhedral patterns in a systematic way; aesthetics and regularity largely come from the structures the patterns have been derived from.

Numerical optimization schemes for computing polyhedral surfaces include the alternating least-squares approach of [Poranne et al. 2013], the local-global projection method of [Bouaziz et al. 2012], the augmented Lagrangian algorithm of [Deng et al. 2015], and the guided projection method of [Tang et al. 2014]. Our computations are based on the latter approach, but are distinct from all the other aforementioned works by the novel use of local affine symmetries as regularizers, which are able to adapt to freeform geometry, and by the study of the curvature-dependent appearance of polyhedral patterns.

Triangle meshes are trivial polyhedral meshes. Regularizing them for face and edge repetitivity is the aim of [Singh and Schaefer 2010] and [Huard et al. 2014]. These meshes also appear in triangle-based point folding structures [Zimmer et al. 2012]. Our symmetry-based regularizers can also be used for triangle mesh optimization.

3 Geometry of Polyhedral Patterns

Creating polyhedral patterns first and foremost poses a theoretical challenge, since we do not possess the knowledge of how such patterns behave in different curvature regions. We have an understanding of planar-quad meshes as given by [Liu et al. 2006]. If the network of polylines that is characteristic of quad meshes follows conjugate directions, it is possible to achieve a mesh with smooth polylines. Unfortunately, this is not possible for any orientation of quads (see auxiliary material for a theoretical proof), and there has been no suggestion for what could be done in this case. An analysis is provided for feasible planar hexagonal tile shapes by [Li et al. 2014]. However, the description is particular for hexagons in principal directions, and the generalization to semi-regular patterns



(a) $(3^4, 6)$ **(b)** $(3, 4, 6, 4)$ **(c)** $(3, 6, 3, 6)$ **(d)** $(3, 12, 12)$ **(e)** $(4, 6, 12)$ **(f)** $(4, 8, 8)$

(g) $(3^4, 6)^*$ **(h)** $(3, 4, 6, 4)^*$ **(i)** $(3, 6, 3, 6)^*$ **(j)** $(3, 12, 12)^*$ **(k)** $(4, 6, 12)^*$ **(l)** $(4, 8, 8)^*$

Figure 5: *Several patterns used in this paper: selected semi-regular patterns (top row; labels correspond to the valences of faces around a vertex) and their duals (bottom row). In our results, patterns (a) - (e) and (g) - (j) are derived from a hex-mesh, patterns (f) and (l) from a quad mesh, and pattern (k) from a triangle mesh.*

is not obvious.

In the following, we provide an analysis of feasible planar tile shapes in different curvature regions, for the general case of semi-regular tilings. In Section 6, we utilize the insights gained from this analysis, to establish a set of symmetries that remain invariant in each curvature region. We consequently use these symmetries as regularizers in our planarization algorithm.

We base our geometric constructions on *semi-regular patterns*, which are tilings that can be derived in the plane by altering either of the three regular tilings: triangle, square, or hexagonal grids. Semi-regular tilings are characterized by several properties: First, the neighborhood of any vertex is perfectly similar to the neighborhood of any other. Second, such tiles constitute an *orthogonal circle pattern*: every tile has a circumcircle, and the dual segments between neighboring tile (circle) centers are orthogonal to the primal edge they share. This property is important when we discuss construction by lifting. We depict a range of semi-regular tilings that we employ in Figure 5. We denote tilings using *vertex configuration* shorthand by numbering the degree of faces around each vertex and using powers for multiples, e.g., a pure hexagonal pattern is 6^3. We use the term "tile" to indicate a single face of the tiling, and either "tiling" or "pattern" to indicate the entire set. We often refer to the dual pattern as the pattern that is made by connecting dual face centers of adjacent primal tiles. Tilings that are not semi-regular exhibit several of these properties, and our consequent optimization results in interesting polyhedral patterns as well.

3.1 Discretization of osculating paraboloids

We construct some explicit embeddings of polyhedral patterns on surfaces to study necessary deformations in tile shapes. Such deformations are the result of fitting planar patterns with given connectivities onto curved surfaces, while constraining each tile to remain planar. Our purpose is to derive the invariants of the required deformations, focusing on symmetries they fix. An understanding of such invariant symmetries serves as a guide to predicting the resulting pattern shapes expected within our optimization process.

We locally approximate the original surface, S, to a second-order in a point, $\mathbf{p} \in S$, with an osculating paraboloid, S_2. Assuming the z direction is parametrized to be in the direction of the normal, the formula defining the paraboloid is $2z = \kappa_1 x^2 + \kappa_2 y^2$, where κ_1, κ_2 are the principal curvatures, and the x and y axes are the respective principal directions. The paraboloids are characterized as either elliptic (both curvatures are nonzero and have the same sign), hyperbolic (different nonzero signs), or cylindrical (one of the curvatures is zero). In case that $\kappa_1 = \kappa_2 = 0$, the osculating paraboloid is a plane, and we do not need to deal further with this trivial case.

We consider discretizations of paraboloids by polyhedral patterns characterized by two properties: First, they are *inscribed*, which means the vertices of the pattern lie exactly on the paraboloid. Second, we have *normal adherence*. Assuming that the supporting plane to the inscribed tile encloses a well-defined small patch of the paraboloid, then there must be a point within the patch whose tangent plane is parallel to the supporting plane. Both properties can be generally relaxed, but it is cogent to study the pattern symmetry and regularity emerging from these most restrictive requirements.

Lifting The analysis we give for tiling surfaces relies on lifting planar tiles onto paraboloids (see Figure 7). We then explore the tile shapes and topologies for which the lifting produces polyhedral patterns (preserving tile planarity). Given the paraboloid S_2, the vertices are lifted (bijectively) by the function $(x, y) \rightarrow \left(x, y, \kappa_1 x^2 + \kappa_2 y^2\right)$. The intersection of the supporting plane of the tile and the paraboloid is a conic, related to the *Dupin indicatrix*. The projection of the conic down to the plane is again a planar conic, endowed with required properties that we next detail. Moreover, the conic on the paraboloid is an *affine image* of the planar tile. See Figure 6 for an example.

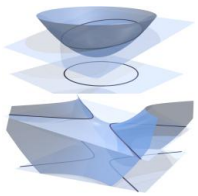


Figure 6: *Lifting. Top: a circle lifted to an ellipse in a rotational paraboloid. Bottom: a hyperbola lifted to a hyperbolic paraboloid.*

Consistent tilings Polyhedral patterns on paraboloids are synonymous with consistent structures that are pivotal to our framework and that govern the deformations induced on tiles for consistent approximation of paraboloids. Assume that there are two neighboring tiles, i, j. Their centers are defined by looking at the centers of the conics on the paraboloid and projecting them down on the tiling

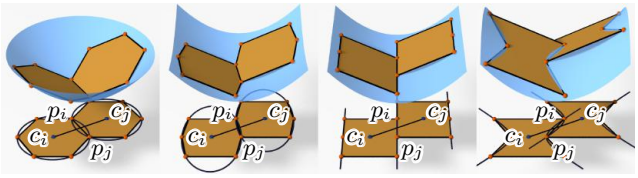


Figure 7: *Lifting consistent tilings to paraboloids. Under the induced metric in the plane, the dual edge $\mathbf{c}_{ij}$ and the primal edge $\mathbf{p}_{ij}$ are conjugate. Both vectors are orthogonal in the rotational paraboloid case (left).*

plane, producing $\mathbf{c}_i, \mathbf{c}_j$. The intersection points between them are the projected common vertices, $\mathbf{p}_i, \mathbf{p}_j$. We define the primal vector $\mathbf{p}_{ij} = \mathbf{p}_j - \mathbf{p}_i$, and the dual vector $\mathbf{c}_{ij}$ similarly. Next, we consider the metric induced by the paraboloid: $\langle \mathbf{a}, \mathbf{b} \rangle := \kappa_1 x_a x_b + \kappa_2 y_a y_b$. The lift of a planar tiling into a paraboloid is polyhedral if and only if:

1. **Conjugacy:** the primal and dual vectors are *conjugate*. i.e., they are orthogonal with respect to the metric ($\langle \mathbf{p}_{ij}, \mathbf{c}_{ij} \rangle = 0$).
2. **Bisection:** The dual edge bisects the primal edge.

We call a tiling that obeys both properties *consistent*. For completeness, we include a proof in the Appendix. Consistency brings about several key consequences:

- If the paraboloid is a rotational-symmetric canonical paraboloid $2z = x^2 + y^2$, then the duals and the primals are in fact also Euclidean orthogonal, as we demand from the original tilings.
- If the paraboloid is cylindrical, then either the dual or the primal must be in the direction of the *ruling* (the direction of zero curvature). That means that the tiling has to comprise strips of faces that are parallel to the ruling direction.
- If the paraboloid is hyperbolic, the dual and the primal can be identical (in *asymptotic* directions). We can potentially produce degenerate and concave tiles in this manner.

3.2 Fitting tiles to paraboloids

Given a planar tiling and a paraboloid of any shape, we next wish to deform the tiling on the plane, such that the lifting produces a polyhedral pattern. This should be done while keeping the shapes of the tiles as symmetric and as regular as possible and with minimal deformation. It is worth noting at this point that we use this construction for a theoretical, rather than a direct algorithmic, purpose; it is a general and arbitrary way to observe which symmetries are invariant within the deformation, and to motivate our use of such symmetries in pattern optimization for general surfaces, locally approximated as paraboloids. We initiate this analysis by considering the canonical rotational paraboloid $S_r : 2z = x^2 + y^2$. We take any semi-regular tiling on the plane. Without any deformation, the circular faces of the semi-regular tiling are then projected into ellipses in the paraboloid, which are planar by definition. We thus obtain a natural polyhedral pattern embedded in S_r for every semi-regular pattern.

Tiling cylindrical paraboloids Without loss of generality, we assume that our cylinder is defined by $S_c : z = \kappa_2 y^2, \kappa_2 > 0$. By the rule of conjugacy, the tiling must comprise strips that are parallel to the ruling direction, $\hat{\mathbf{y}}$, to consistently tile the cylinder. In light of this, and opting to deform the tiling as little and as symmetrically as possible, we do the following: decompose the tiling into strips of faces that are parallel to the rulings. The lines between dual

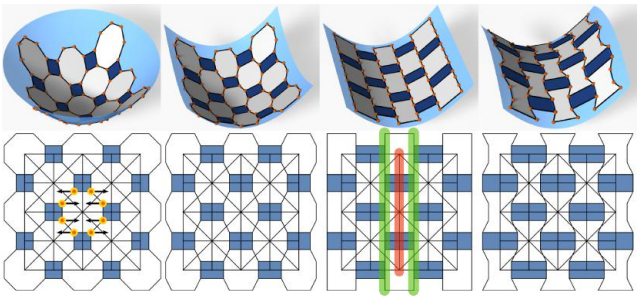


Figure 8: *Deforming and lifting. Top: the paraboloid tilings. Bottom: top view of the tiling. Left to right: original (fit to the canonical paraboloid), anisotropic elliptic, cylindrical, and hyperbolic.*

conic centers of the same strips are denoted as *dual rulings*, and the sequence of intersection edges bounding two strips are denoted as *primal rulings*. Next, fix all the tile (dual) centers and deform all (primal) vertices orthogonally to the ruling until they are on a parallel primal ruling. The actual position of the primal ruling is set as the ruling that is closest to the original primal vertices (see Figure 8 for a depiction of this process). Denote the original position of any primal vertex as $\mathbf{p}_i$, and the deformation vector for a cylindrical pattern as $\mathbf{u}_{c,i}$. Then, $\hat{\mathbf{y}} \cdot \mathbf{u}_{c,i} = 0$.

Tiling elliptic and hyperbolic paraboloids To unify our deformation setting and make it fit all types of paraboloids, we rephrase our construction for a cylindrical paraboloid in a local and continuous manner: we deform the primal edges of an initial tiling so that the (constant) dual and the primal are conjugate according to the induced metric, and the deformation is done according to the choice of strip decomposition, as in the cylindrical case. Suppose again that the chosen direction of the dual strip is $\hat{\mathbf{y}}$. We then need to find the deformation vector $\mathbf{u}_i$ of vertex $\mathbf{p}_i$ so that for each primal edge $\mathbf{p}_{ij}$ and dual edge $\mathbf{c}_{ij}$ we get $\langle \mathbf{p}_j + \mathbf{u}_j - \mathbf{p}_i - \mathbf{u}_i, \mathbf{c}_{ij} \rangle = 0$, according to the metric. Furthermore, we constrain $\mathbf{u}_i = -\mathbf{u}_j$ for symmetry. Since $\mathbf{u}_i = -\mathbf{u}_j$ are orthogonal to $\hat{\mathbf{y}}$, it is straightforward to compute the actual deformation.

Discussion Our deformation process is obviously invariant to scale. More accurately, it only pertains to the ratio of κ_1 and κ_2 and not their sizes. In addition, it is evident that tiles with more than 4 vertices must become non-convex in order to be inscribed in negatively curved regions; this is expected, since the conic in which the tile is inscribed is a hyperbola.

The deformation orthogonal to the chosen strip direction is the only one possible for tilings such as 6^3. However, by constraining the consistency, and by conforming to the aesthetic request that the repeating faces of the same type must stay congruent, some tilings may in fact allow more degrees of freedom in deformation possibilities. This works only to our advantage.

Violating consistency The deformations we defined maintain conjugacy and bisection for most semi-regular patterns, but not for some. For example, consider the $(4, 6, 12)$ example in Figure 9: by fixing all the dual vertices, our deformation would violate bisection between some of the quads and the 12-sided faces. This is caused by the special structure of the $(4, 6, 12)$ pattern, for which the dual centers of the hexes and the oblique quads cannot conform to straight rulings should they stay fixed. Our correction is simple: allow the centers of the hexes to deform as well, so that they line up with the (fixed) centers of the oblique quads.

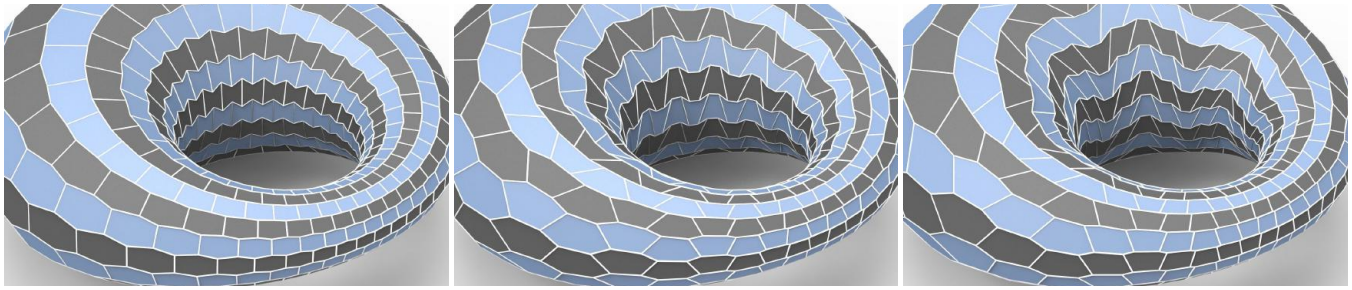


Figure 11: *Different strip decompositions increase the available types of polyhedral patterns. With the three strip decompositions shown in Figure 10, we obtain hexagon patterns approximating a cyclide with different appearances.*

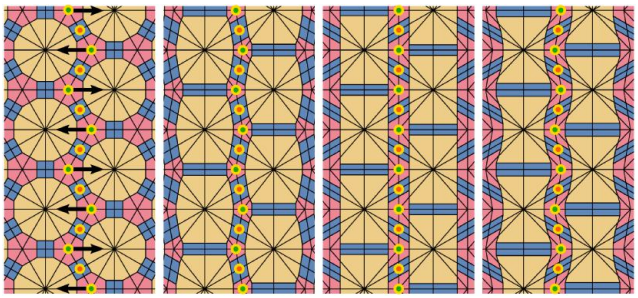


Figure 9: *The dual centers of the* $(4, 6, 12)$ *pattern do not correspond to possible rulings if fixed upon deformation. Allowing the hex (pink) centers to deform fixes this problem.*

3.3 Strip decompositions

Our explicit construction provides a canonical way to approximate paraboloids, by relying on a single possible choice for defining strips. In the following, we explore other possible constructed solutions, by choosing different alignments, corresponding to decomposing the patterns into different strips.

Decomposable patterns Essentially, strip decomposition is a combinatorial refinement of the original pattern. A feasible strip decomposition is a collection of disjoint dual strings (trees with 2-valence nodes; see Figure 10 for examples). Since a strip decomposition assigns primal vertices to dual vertices, it is actually a strip decomposition of the dual pattern as well. Regular quads, hexagons, $(3^4, 6)$, $(3, 4, 6, 4)$, $(4, 6, 12)$, $(4, 8, 8)$ and their dual patterns have infinitely many strip decompositions. However, patterns such as $(3, 6, 3, 6)$, $(3, 12, 12)$ and their duals cannot be decomposed to strips by definition.

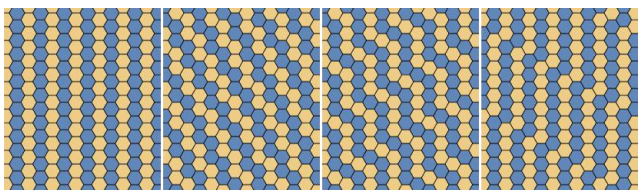


Figure 10: *Different strip decompositions for regular hexagons. The three decompositions from the left correspond to the ones shown in Figure 11. In addition, the transformation corresponding to the decomposition second from the left is shown in Figure 12.*

Deformations and symmetries The practical meaning of choosing strips is to contrast a chosen strip direction with the principal directions of the paraboloid. Choosing a strip means choosing a dual axis for every consequent pair of faces, and forcing the primal vertices to move in directions that are orthogonal to the strip axis. In other words, we constrain a *plane of symmetry* that is orthogonal to the tile, and passes through its center.

However, as we explain above, the principal directions may mandate (e.g., if they are rulings), that the surface forms lines of primal vertices along such rulings. At any rate, the tile must also be inscribed to a conic of the same nature. The *canonical* choice of strips is where the dual axis is aligned with the rulings (see Figure 10 left). However, other choices may produce interesting patterns due to the mismatch between the constrained symmetry and the rulings (see Figure 12), and they may potentially form invalid configurations. Such configurations arise when the strips are along the asymptotic directions of the surface, i.e., where the dual direction is self conjugate.

General patterns Not all patterns can be decomposed to strips. For instance, the tri-hex pattern $(3, 6, 3, 6)$ cannot be decomposed. Such patterns cannot therefore comply to the normal adherence and cannot be made consistent by deformation. However, as our algorithm requires consistency only for the theoretical analysis, we still utilize these patterns in practice, just without any guarantees. Figures 17 and 23 provide examples.

3.4 Regularizers Motivated by Symmetries

Ideally, we would like to achieve the described symmetric and planar tile shapes for meshes initially tiled with semi-regular convex patterns. However, general meshes are not paraboloids, and they have a variety of strip decompositions and varying curvature regions. Not wanting to be particular for every pattern, we instead opt for the most general way to make any type of semi-regular pattern deform properly. Our point is to utilize what *remains invariant* under curvature-based tile deformations, rather then what *deforms*. There-

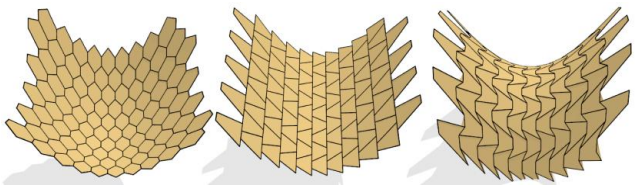


Figure 12: *Transformation of the regular hexagon pattern from a rotational paraboloid (left) via a parabolic cylinder (middle) to a hyperbolic paraboloid (right) with the strip decomposition shown in Figure 10, second from left.*

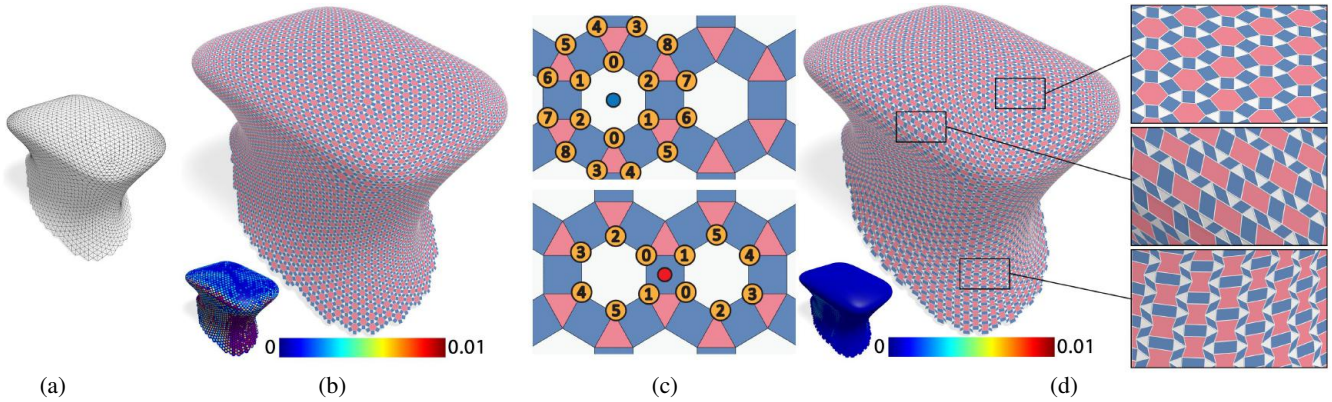


Figure 13: *Framework overview: a) for an initial triangle, quad, or hex mesh, we can generate a pattern mesh using simple geometric rules. b) The initial pattern mesh might already be aesthetically pleasing, but the faces are typically not planar. c) A regularizer can be configured by specifying symmetries that should be preserved in the pattern. In this case, face symmetries are chosen. Corresponding vertex pairs are shown using the same number and the symmetry centers are shown in blue and red. d) Finally, the optimization generates a mesh with planar faces. The most interesting aspect of polyhedral patterns is that most of them have to transform so that they look different in regions of positive, zero, and negative Gaussian curvature (see insets).*

fore, we identify invariant symmetries of tiles and then regularize the mesh in our planarization process to maintain them. The symmetries that we identify include reflection through axes and through planes as well as reflections through the centers of tiles or edges. It is straightforward to check that such symmetries are general enough to contain the deformations we describe here. The symmetries are described in greater detail in Section 6.

4 Overview and User Interaction

Our framework comprises four stages, shown in Figure 13.

Pre-processing Initial meshes are generated using triangular-, quad-, or hex-based remeshing techniques in a separate program, according to the desired pattern (see Figure 5 for a description). Many patterns are initialized using the hex-based remeshing approach proposed by Vaxman and Ben-Chen [2015] with their planarity optimization omitted. We therefore have two input meshes in our system: A finely-tessellated triangle mesh to define the reference surface and a coarser (non-planar) remeshed triangle, quad, or hex mesh.

Pattern generation The user can transform the initial coarse mesh into a pattern mesh by selecting from a list of pre-defined patterns. The transformation is implemented using a sequence of geometric rules, e.g., subdivision rules. The implementation of such rules is fairly straightforward and follows the framework proposed by Akleman et al. [2005].

Symmetry configuration The user can then specify a desired strip decomposition and configure the regularizer by assigning symmetries. We offer axial symmetries with respect to an axis passing through a vertex, an edge midpoint, or a face barycenter. We also offer reflective symmetries with respect to a plane, e.g., a plane passing through an edge. The user specifies the symmetry assignment for one or more elements, and the system propagates the assignment over the whole mesh according to the strip decomposition. To guide the user in his/her selection, we provide a list of suggested symmetries. The suggestions are generated by mapping each strip decomposition of each pattern to a cylinder and then observing what symmetries are feasible (see Sec. 3).

Symmetry optimization Our algorithm optimizes the pattern for planarity and aesthetics (using the regularizer configured in the previous step) through non-linear optimization. The details for the optimization framework are presented in Section 5, and those for the symmetry regularizers are given in Section 6.

5 Optimization Framework

We next describe the regularity-based planarity optimization framework that our work builds upon. The inputs are a reference surface, S, given as a triangle mesh, and an initial polygonal mesh with vertices, $\mathbf{v}_i$, that approximates the reference surface. The goal is to optimize the initial polygonal mesh, $M = (V, E, F)$, according to three terms: the planarity of the faces, the closeness to the reference surface, and the regularity of the mesh. We rely on existing methods (described in this section) to formulate planarity and closeness terms. The regularity terms are our contribution.

Variables We denote the vertex coordinates of M as $\mathbf{v}_i,\ i \in V$, and the unit face normals as $\mathbf{n}_k,\ k \in F$. Vertices are not constrained to lie on the reference surface, S, exactly. The closest point on S for a vertex $\mathbf{v}_i$ is $\mathbf{v}_i^*$ with corresponding normal $\mathbf{n}_i^*$ (see Fig. 14).

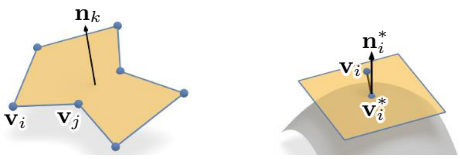


Figure 14: *Notation: faces (left), closest point projection (right)*

Problem formulation The objective function we minimize is

$$E(\mathbf{v}_i) = \lambda_1 E_{plan} + \lambda_2 E_{close} + \sum_j \mu_j E_{reg}^j. \qquad (1)$$

Following the reasoning of [Tang et al. 2014], we set up a system with energies that are at most quartic, which entails soft constraints that are at most quadratic, since this formulation is easy to optimize using a standard regularized Gauss-Newton algorithm.

Planarity The planarity constraint is necessary for all non-triangular faces. We adapt the formulation of [Tang et al. 2014] and express E_{plan} as

$$E_{plan} = \sum_{k \in F} \sum_{(i,j) \in E(f_k)} ((\mathbf{v}_i - \mathbf{v}_j) \cdot \mathbf{n}_k)^2 + \sum_k (\mathbf{n}_k \cdot \mathbf{n}_k - 1)^2, \quad (2)$$

which is zero if all face edges are orthogonal to a unit length normal.

Closeness The closeness constraint of a vertex, $\mathbf{v}_i$, to a reference surface is modeled by requiring $\mathbf{v}_i$ to move only on the tangent plane associated with its closest point, $\mathbf{v}_i^*$, on the reference surface, S:

$$E_{close} = \sum_{\mathbf{v}_i \in V} ((\mathbf{v}_i - \mathbf{v}_i^*) \cdot \mathbf{n}_i^*)^2. \quad (3)$$

As shown in Figure 14, $\mathbf{n}_i^*$ is the normal of the tangent plane at $\mathbf{v}_i^*$, and it is kept constant in every iteration. Alternatively, for coarse and inconsistent tilings, we may use closeness of face barycenters instead to relax this constraint (see Figures 18 and 29, left).

Previous iteration We add a term in each iteration that dampens the optimization for stability by closeness to the previous iteration:

$$E_{prev} = \beta \sum_{\mathbf{v}_i \in V} \left| \mathbf{v}_i^m - \mathbf{v}_i^{m-1} \right|^2 + \beta \sum_{\mathbf{n}_i \in F} \left| \mathbf{n}_i^m - \mathbf{n}_i^{m-1} \right|^2, \quad (4)$$

where m denotes the number of an iteration and v_i^m is the value of v_i at iteration m. We use $\beta = 10^{-6}$ in all our examples.

6 Regularization with Affine Symmetries

We next define the invariant symmetries of deforming patterns (see Section 3) and how we utilize them in practice to regularize patterns undergoing deformations through the planarization process. Generally speaking, there are several ways to represent feasible regularities, such as enforcing specific angles, polyline smoothness of selected sequences of non-adjacent vertices, ratios of edge lengths, and more. We choose to use local affine symmetries as described in the following, because they are simple, sparse, local and linear (e.g., compared with angle-based formulations), and this is important for computational efficiency.

We describe the practical implementation of various symmetry regularizers and adapt them to the discrete surface by two approaches: affine symmetries in space and in a tangential projection. Each approach has different merits and shortcomings.

6.1 Affine symmetries

Affine symmetries can be defined with respect to either an axis or a plane. We can distinguish four different generators of symmetries: vertices, faces, edge midpoints, and edges. Vertices, faces, and edge midpoints generate symmetries with respect to an axis and edges generate symmetries with respect to a plane.

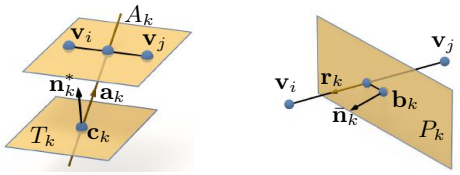


Figure 15: *Affine reflection in an axis (left) and in a plane (right)*

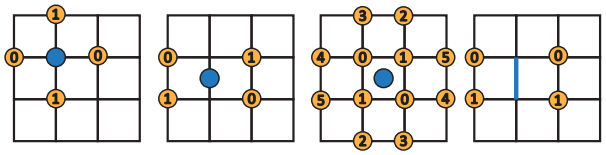


Figure 16: *Left to right: symmetry with respect to a vertex, an edge midpoint, a face barycenter, and an edge.*

Axial symmetries An affine reflection in an axis, A_k, requires the additional prescription of a reference plane, T_k (not parallel to A_k; see Fig. 15, left). Then, a pair of vertices $\mathbf{v}_i$ and $\mathbf{v}_j$ is symmetric with respect to A_k if the midpoint between $\mathbf{v}_i$ and $\mathbf{v}_j$ lies on A_k and the vector $\mathbf{v}_i - \mathbf{v}_j$ is parallel to T_k. Let A_k be defined by a direction vector $\mathbf{a}_k$ and a point $\mathbf{c}_k$, and let $\mathbf{n}_k^*$ be a normal vector of T_k. Then, the axial symmetry regularizer E_{reg}^1 is encoded as follows:

$$\sum_{(i,j,k)} ((\mathbf{v}_i + \mathbf{v}_j)/2 - (\mathbf{c}_k + \lambda_{kl}\mathbf{a}_k))^2 + ((\mathbf{v}_i - \mathbf{v}_j) \cdot \mathbf{n}_k^*)^2 \quad (5)$$

The triplets (i, j, k) are chosen according to the user-assigned symmetries. That is, $\mathbf{v}_i$ and $\mathbf{v}_j$ are selected to have affine symmetry with respect to A_k. Furthermore, $\mathbf{a}_k$ and λ_{kl} are considered as additional variables in our optimization. The axis point $\mathbf{c}_k$ is a vertex, an edge midpoint, or a face barycenter, and thus it is a linear combination of vertex coordinates. The $\mathbf{n}_k^*$ variables can be either considered as additional variables, or approximated at the beginning of each iteration as the normal of S at the closest point to $\mathbf{c}_k$. We do the latter. Symmetry is applied to local neighborhoods as illustrated in Fig. 16. Note that the symmetry of a planar face with respect to its barycenter does not require an axis. $\mathbf{n}_k^* = \mathbf{a}_k$ models a Euclidean reflection in A_k and is therefore suitable to enforce Euclidean symmetries.

Plane-reflective symmetries Fig. 15 (right) presents an affine reflection with respect to a plane, P_k (through point $\mathbf{b}_k$ and with normal vector $\bar{\mathbf{n}}_k$), in the direction $\mathbf{r}_k$. Plane-reflective symmetry of $\mathbf{v}_i$ and $\mathbf{v}_j$ requires their midpoint to be located on P_k, and the vector $\mathbf{v}_i - \mathbf{v}_j$ to be parallel to $\mathbf{r}_k$. We encode this requirement in the regularizer, E_{reg}^2, as follows:

$$\sum_{(i,j,k)} (((\mathbf{v}_i + \mathbf{v}_j)/2 - \mathbf{b}_k) \cdot \bar{\mathbf{n}}_k)^2 + ((\mathbf{v}_i - \mathbf{v}_j) - \lambda_{kl}\mathbf{r}_k)^2. \quad (6)$$

We use $\mathbf{r}_k$ for each reflection plane and the scale variable λ_{kl} as additional variables. The plane P_k is commonly the bisection plane of two adjacent faces in a polyhedral mesh, and so $\mathbf{b}_k$ can be the midpoint of the common edge of two adjacent faces. The normal, $\bar{\mathbf{n}}_k$, is then pre-estimated in each iteration. With $\mathbf{r}_k = \bar{\mathbf{n}}_k$, we obtain a Euclidean reflection.

Symmetry centers In Fig. 16, as exemplified on a quad mesh, the three leftmost images show point symmetries in 2D, equivalent to 3D axial symmetries. The blue dot represents the symmetry center. Each pair of symmetric points is labelled the same (orange dots). Their symmetry centers are located at a vertex, an edge midpoint or a face barycenter, and their symmetries are denoted accordingly. The rightmost edge symmetry is equivalent to a 3D planar symmetry.

6.2 Symmetry in a tangential projection

A relaxed version of affine axial symmetry is symmetry in a projection parallel to a certain direction (the image plane of the projection does not matter). To achieve it, we simply discard the second part of

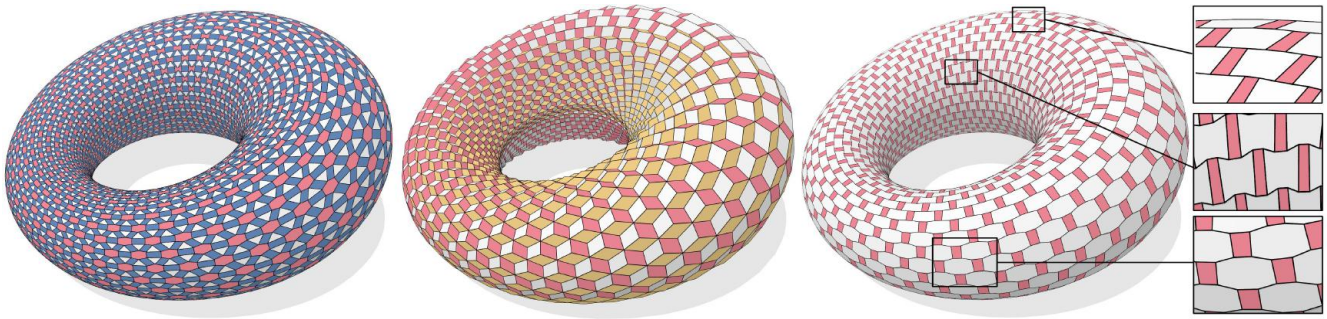


Figure 17: *Semi-regular patterns on a Dupin cyclide. Left: A* $(3,4,6,4)$ *pattern using face symmetries. Middle: A* $(3,6,3,6)^*$ *pattern using symmetries with respect to an edge. Right: A* $(4,8,8)$ *pattern using face symmetries.*

Equation 5. We use the normal at the closest point, $\mathbf{c}_k^*$, to $\mathbf{c}_k$ as the projection direction, and thus enforce a symmetry that is relative to the tangent plane of S at $\mathbf{c}_k^*$. This yields the regularizer E_{reg}^3:

$$\sum_{(i,j,k)} ((\mathbf{v}_i + \mathbf{v}_j)/2 - (\mathbf{c}_k + \lambda_{kl}\mathbf{a}_k))^2. \tag{7}$$

By approximating $\mathbf{a}_k$ with $\mathbf{n}_k^*$, this expression can be simplified to the following equivalent formulation:

$$\sum_{(i,j,k,l)} (((\mathbf{v}_i + \mathbf{v}_j)/2 - \mathbf{c}_k) \cdot \mathbf{t}_{kl})^2. \tag{8}$$

In addition, we need to sum over two orthogonal directions, $\mathbf{t}_{k1}$ and $\mathbf{t}_{k2}$. Both directions are orthogonal to $\mathbf{n}_k^*$, and they are estimated at the beginning of each iteration.

6.3 Avoiding self-intersection

The proposed symmetry regularizers cannot prevent self-intersections within the pattern. To counter that, we introduce an additional regularizer, E_{reg}^4, assuming that a line segment connecting the face barycenter to a vertex is within the corresponding face:

$$\sum_{\substack{k\in F \\ 0\le i<|f_k|}} ((\hat{\mathbf{v}}_k^i - \mathbf{c}_k) \times (\hat{\mathbf{v}}_k^{i+1} - \mathbf{c}_k) \cdot \mathbf{n}_k - \nu_{ki}^2)^2. \tag{9}$$

We assume that the faces are consistently oriented with vertices $(\hat{\mathbf{v}}_k^0, \hat{\mathbf{v}}_k^1, \ldots, \hat{\mathbf{v}}_k^{|f_k|-1})$, where indices in the sum are taken modulo the face valence $|f_k|$. The face normals, $\mathbf{n}_k$, and barycenters, $\mathbf{c}_k$, are evaluated prior to each iteration, and considered as constants. We introduce ν_{ki} as *slack variables* to encode the inequality requirements that the vectorial areas of $\mathbf{c}_k\hat{\mathbf{v}}_k^i\hat{\mathbf{v}}_k^{i+1}$ are aligned with $\mathbf{n}_k$.

6.4 Edge length regularization

To avoid short edges, we regularize selected length differences of adjacent edges. For example, if $\mathbf{e}_i$ and $\mathbf{e}_j$ are the edge vectors of two neighboring edges, we regulate the ratio of their lengths into a given interval, and E_{reg}^5 is defined as:

$$\sum_{(i,j)} (\|\mathbf{e}_i\|^2 - r^2\|\mathbf{e}_j\|^2 - \mu_{ij}^2)^2 + (\|\mathbf{e}_j\|^2 - r^2\|\mathbf{e}_i\|^2 - \mu_{ji}^2)^2, \tag{10}$$

where the summation is over the pairs of edges chosen based on the used strip decomposition and r is set to 0.8 in our implementation. We can require a lower bound, l_{min}, on edge lengths by E_{reg}^6:

$$\sum_{i\in E} (\|\mathbf{e}_i\|^2 - l_{min}^2 - \gamma_i^2)^2. \tag{11}$$

μ_{ij} and γ_i are slack variables for the inequality requirements.

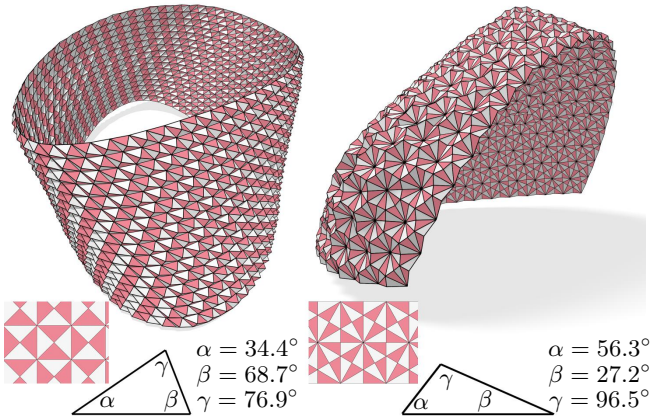


Figure 18: *Triangle meshes constructed by a single type of triangle for the* $(4,8,8)^*$ *pattern (left) and the* $(4,6,12)^*$ *pattern (right) using symmetries with respect to an edge.*

7 Results

We present several results and discuss parameters, planarity, running time, failure cases, and comparison to related work.

Models and patterns We generate results for a set of selected surfaces showing multiple patterns per surface (see Figures 17, 23, and 24) where all patterns use the simplest strip decomposition. Throughout the paper, we select the surfaces to highlight the behavior of our regularizer in different situations. We select models to include regions of positive and negative Gaussian curvature, as well as interesting transition regions between them. We exemplify patterns on both open and closed surfaces, as well as surfaces with topological holes. We show a wood construction of an interior cladding of an architectural model in Figure 2. Finally, our method is used to generate rough patterns consisting of identical triangles (see Figure 18) and a $(3,4,6,4)$ pattern on a non-architectural model (see Figure 19). Note that our algorithm typically targets architectural models with moderate curvature variations and a sparse set of singularities. This model is therefore mainly depicted to demonstrate robustness.

Strip decompositions We demonstrate a pattern on a surface with different strip decompositions (see Figures 1 and 30). Different symmetries are used to accommodate each decomposition. Combining different strips and symmetries leads to a large variety of aesthetic results from a single basic pattern (see Figure 29).

Figure 19: *We show a* $(3, 4, 6, 4)$ *pattern on the Moomoo model with 57 singularities. This is the most complicated model shown in the paper due to the high number of singularities (hexagons replaced by septagons and pentagons) and the high curvature variations.*

Planarity All our models are successfully planarized. For assessment, we measure face planarity as the maximum distance between a vertex and a regression plane computed using PCA, normalized by dividing by the average edge length in the model. Our tolerance for this measure is under 10^{-2}. We illustrate one example in Figure 13 where we show how the Soumaya museum model is planarized. Before the optimization, many faces of the model are considerably non-planar. The optimization is nevertheless successful.

Parameters The main parameters stem from the configuration of the regularizer, i.e., the selected symmetries and strip decompositions. The weights for the different terms in the optimization vary slightly per pattern. In the optimization, we use the default parameters of 1.0 for face planarity, 0.1 for the closeness to the reference surface, and 0.01 for the symmetry-based regularizers. This choice is good for most examples. For example, the $(3^4, 6)$ pattern in Figure 24, right and the $(3, 4, 6, 4)$ pattern in Figure 17, left use these parameters without change.

Alternatively, the user can adjust the parameters to trade off regularity for stricter planarity or closeness to the reference surface; see Figure 20 for example. For the $(4, 6, 12)$ patterns in Figure 23 and 24, we set the planarity to 5.0 and 10.0, respectively, while setting the closeness and symmetry parameters to the default ones. For the remaining terms, we advise that the edge-length parameter be equated with the symmetry regularization parameter, and the self-intersection avoidance parameter with the planarity parameter.

Implementation details and running times We implemented our framework in C++ using OpenMesh [Botsch et al. 2002] for the mesh data structures and TAUCS [Krieg et al. 2003] as the library for sparse linear solvers. The running times for our examples are typically under one minute with an Intel Xeon X5550 2.67GHz processor. For example, the Soumaya model in Figure 13 has 7034 vertices and took 43 seconds to optimize. Smaller examples are much faster. For example, a $(4, 6, 12)$ pattern on an HP surface with

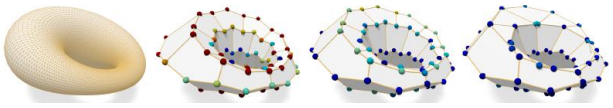


Figure 20: *A planarized cyclide model. Hotter vertex colors indicate greater distances to the reference surface (left). A coarse input mesh can be successfully planarized and regularized at the cost of lower fidelity to the reference surface (second from left). Forcing closeness to the reference surface sacrifices regularity (second from right) and may cause edge degeneracies (right).*

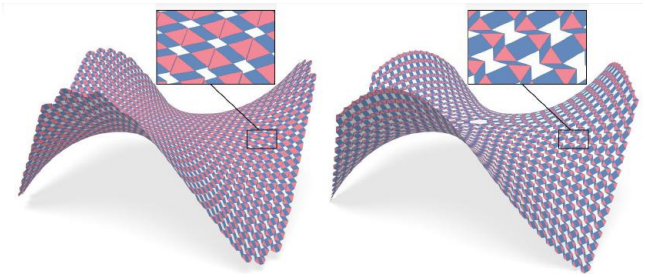


Figure 21: *Failure case on a monkey saddle: due to the initialization used on the left, the pattern* $(3, 4, 6, 4)$ *degenerates as some of the quads collapse to lines. With a different initialization that is better aligned with the principal curvature directions, the pattern can be mapped correctly (right).*

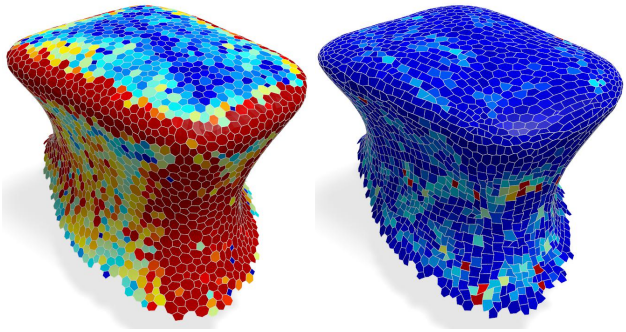


Figure 22: *A failure case due to improper initialization. With a hex-dominant mesh dualized from of an arbitrary triangle mesh (left), the computed result (right) cannot achieve regularity or planarity, because there are too many singularities, leading to a over-fragmented strip decomposition.*

432 vertices took 0.53 seconds to optimize.

Failure cases and limitations Our framework is sensitive to the triangle, quad or hex mesh that is used as input. A poor initialization leads to poor results. In Figure 21, we contrast the results of a poorly initialized optimization with a good initialization on the monkey saddle. In Figure 22, we show a poor initialization of the Soumaya model with many singularities in comparison with the example demonstrated above in Figure 13. In addition, we depend on the quality of existing code for the creation of the initial hex or quad patterns on the surface.

Coarse meshes Our algorithm can create adequate results on coarse meshes as well. However, we observe that coarse meshes can typically only be planarized and regularized when sacrificing the closeness to the reference surface, as demonstrated in Figure 20.

Comparison to PQ meshing Our regularizers provide more degrees of freedom than does the traditional regularizer based on polyline fairness [Tang et al. 2014]. Therefore, our planarization is less sensitive to PQ meshes that are not initialized according to conjugate directions. We show examples in Figures 4, 27, and 29. A detailed analysis of the difference is provided in the additional materials.

Comparison to PH meshing Li et al. [2014] proposed a regularizer for planar hex meshes, meshing the positively and negatively

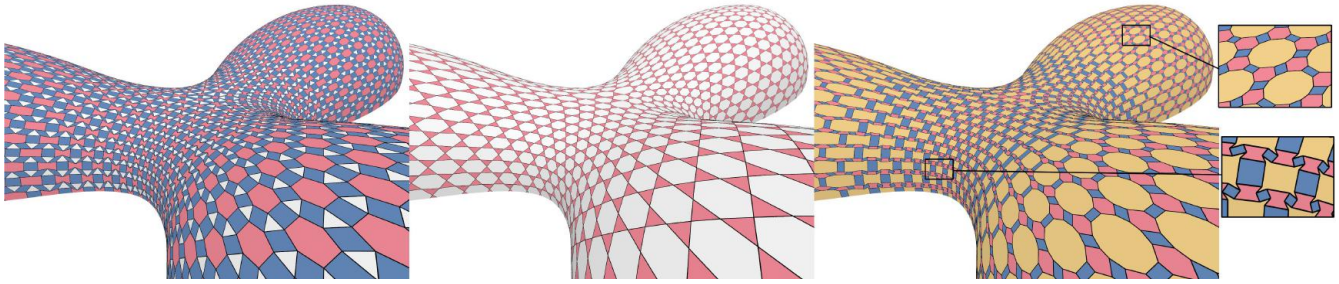


Figure 23: *Left: A* $(3, 4, 6, 4)$ *pattern using face symmetries. Middle: A* $(3, 6, 3, 6)$ *pattern using vertex symmetries. Right: A* $(4, 6, 12)$ *pattern using face symmetries.*

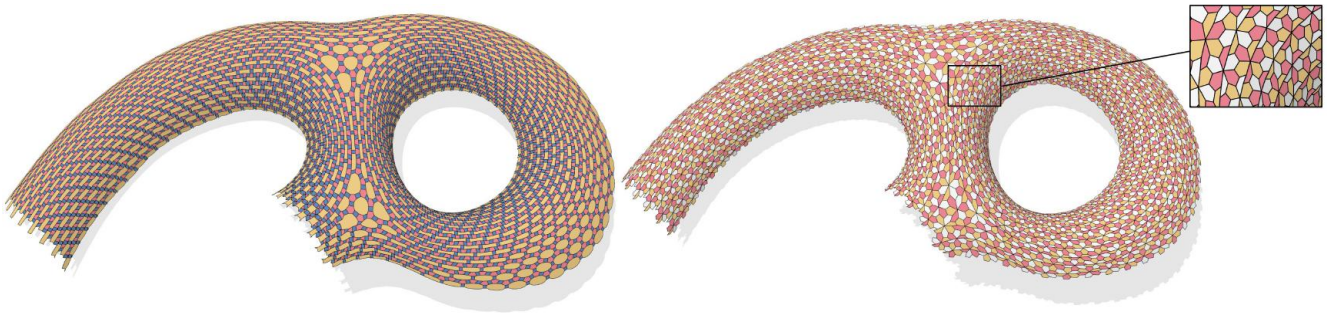


Figure 24: *Semi-regular patterns on an architectural six shape. Left: A* $(4, 6, 12)$ *pattern using face symmetries. A* $(3^4, 6)^*$ *pattern using vertex symmetries. Note how prominent feature lines form automatically due to the regularization.*

curved regions separately. However, they do not propose a specific solution for the transition region and cannot automatically assign which regularizer to use. This may lead to artifacts in the transition region (see Figure 25) and to possible failures in the planarization (Figure 26). Our solution can produce significantly better results. It also caters to non-canonical strip decomposition. However, we note that Li et al. [2014] propose a complete framework for PH remeshing, while our paper focuses only on the regularizer used after the generation of an initial mesh layout.

Comparison to ad-hoc regularizers A large variety of regularizers has been proposed in other contexts. When using ad-hoc regularizers, typical problems occur, depending on how the regularizer is weighted. On the one hand, using a high weight leads to a mesh that is visually pleasing, but not polyhedral. On the other hand, using a low weight leads to a planar mesh that is highly irregular, including degeneracies like self-intersections. There is no effective weight that can achieve both planarity and regularity simultaneously. Figures 27 and 28 present examples.

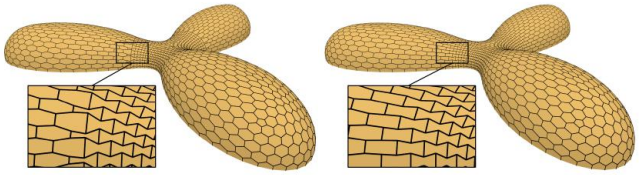


Figure 25: *Comparison of Phex mesh aesthetics: with the same initialization, the regularizer of [Li et al. 2014] generates the left mesh, while our approach leads to more natural transitions (right).*

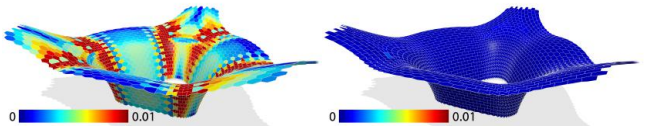


Figure 26: *Comparison of Phex mesh planarity. Left: [Li et al. 2014]. Right: our method achieves better planarity.*

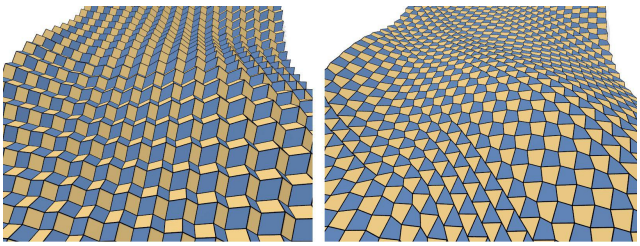


Figure 29: *Different patterns generated by different symmetries. The pattern on the left uses face symmetries; the one on the right is based on edge midpoint symmetries.*

8 Conclusions

We consider the design and optimization of polyhedral patterns, i.e. patterns of planar polygonal faces, on freeform surfaces. Our contributions are the description of a novel class of regularizers based on affine symmetries and a theoretical analysis of polyhedral patterns. In future work, we plan to study mixed patterns and their transition regions, volumetric patterns such as frame structures for support in architectural applications, folding patterns, and time-varying polyhedral patterns for shading systems.

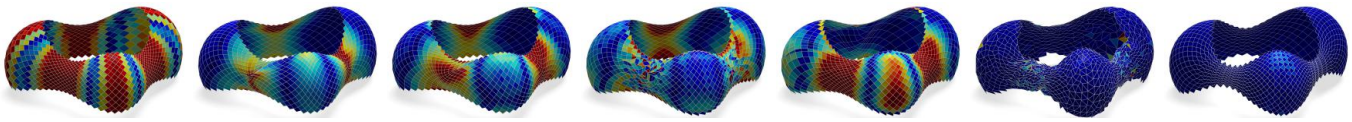


Figure 27: *Mesh planarity. From left to right: initialization, polyline fairness, Laplacian, edge length, angles, face area, ours.*

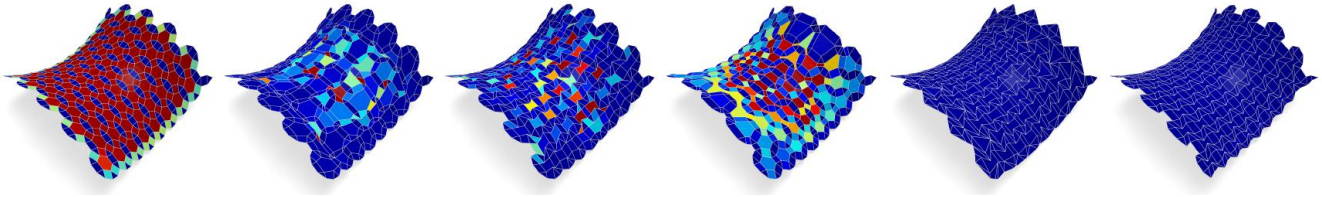


Figure 28: *Mesh planarity. From left to right: initialization, Laplacian, edge length, angles, face area, ours.*

Acknowledgments

We thank the anonymous reviewers for their insightful comments and suggestions for improving the paper. This research was supported by the Visual Computing Center (VCC) at KAUST and by the Austrian Science Fund (FWF) via grants P23735-N13 and I706-N26 (DFG-Collaborative Research Center, TRR 109, Discretization in Geometry and Dynamics). Caigui Jiang and Chengcheng Tang were supported by KAUST baseline funding.

References

ABAS, S. J., SALMAN, A. S., MOUSTAFA, A., AND ATIYAH, M. 1995. *Symmetries of Islamic geometrical patterns*, vol. 3. World Scientific.

AKLEMAN, E., SRINIVASAN, V., AND MANDAL, E. 2005. Remeshing schemes for semi-regular tilings. In *Proc. Inter. Conf. on Shape Modeling and Applications*, 44–50.

BOBENKO, A., AND SURIS, YU. 2008. *Discrete differential geometry: Integrable Structure*. American Math. Soc.

BOTSCH, M., STEINBERG, S., BISCHOFF, S., AND KOBBELT, L., 2002. Openmesh - a generic and efficient polygon mesh data structure.

BOUAZIZ, S., SCHWARTZBURG, Y., WEISE, T., AND PAULY, M. 2012. Shaping discrete geometry with projections. *Computer Grapics Forum 31*, 1657–1667.

COHEN-STEINER, D., ALLIEZ, P., AND DESBRUN, M. 2004. Variational shape approximation. *ACM Trans. Graphics 23*, 3, 905–914.

DENG, B., BOUAZIZ, S., DEUSS, M., KASPAR, A., SCHWARTZBURG, Y., AND PAULY, M. 2015. Interactive design exploration for constrained meshes. *Computer-Aided Design 61*, 13–23.

HUARD, M., BOMPAS, P., AND EIGENSATZ, M. 2014. Planar panelization with extreme repetition. In *Advances in Architectural Geometry 2014*, P. Block et al., Eds. Springer.

JIANG, C., WANG, J., WALLNER, J., AND POTTMANN, H. 2014. Freeform honeycomb structures. *Comput. Graph. Forum 33*, 5, 185–194.

KRIEG, O. D., SCHWINN, T., MENGES, A., LI, J.-M., KNIPPERS, J., SCHMITT, A., AND SCHWIEGER, V. 2003. Taucs. In *A Library of Sparse Linear Solvers*. TAU.

KRIEG, O. D., SCHWINN, T., MENGES, A., LI, J.-M., KNIPPERS, J., SCHMITT, A., AND SCHWIEGER, V. 2014. Biomimetic lightweight timber plate shells: Computational integration of robotic fabrication, architectural geometry and structural design. In *Advances in Architectural Geometry*. Springer, 109–125.

LI, Y., LIU, Y., AND WANG, W. 2014. Planar hexagonal meshing for architecture. *IEEE Trans. Vis. Comp. Graphics 21*, 95–106.

LIU, Y., POTTMANN, H., WALLNER, J., YANG, Y.-L., AND WANG, W. 2006. Geometric modeling with conical meshes and developable surfaces. *ACM Trans. Graph. 25*, 3, 681–689.

LIU, Y., XU, W., WANG, J., ZHU, L., GUO, B., CHEN, F., AND WANG, G. 2011. General planar quadrilateral mesh design using conjugate direction field. *ACM Trans. Graph. 30*, #140, 1–10.

LU, P. J., AND STEINHARDT, P. J. 2007. Decagonal and quasi-crystalline tilings in medieval islamic architecture. *Science 315*, 5815, 1106–1110.

PORANNE, R., OVREIU, E., AND GOTSMAN, C. 2013. Interactive planarization and optimization of 3D meshes. *Comput. Graph. Forum 32*, 1, 152–163.

SAUER, R. 1970. *Differenzengeometrie*. Springer.

SCHIFTNER, A., HÖBINGER, M., WALLNER, J., AND POTTMANN, H. 2009. Packing circles and spheres on surfaces. *ACM Trans. Graph. 28*, 5, #139,1–8.

SINGH, M., AND SCHAEFER, S. 2010. Triangle surfaces with discrete equivalence classes. *ACM Trans. Graph. 29*, #46,1–7.

TANG, C., SUN, X., GOMES, A., WALLNER, J., AND POTTMANN, H. 2014. Form-finding with polyhedral meshes made simple. *ACM Trans. Graphics 33*, 4.

VAXMAN, A., AND BEN-CHEN, M. 2015. Dupin meshing: A parameterization approach to planar hex-dominant meshing. Tech. Rep. CS-2015-01 (CS series), Technion.

ZADRAVEC, M., SCHIFTNER, A., AND WALLNER, J. 2010. Designing quad-dominant meshes with planar faces. *Comput. Graph. Forum 29*, 5, 1671–1679.

ZIMMER, H., CAMPEN, M., BOMMES, D., AND KOBBELT, L. 2012. Rationalization of triangle-based point-folding structures. *Comput. Graph. Forum 31*, 611–620.

ZIMMER, H. 2014. *Optimization of 3D models for fabrication*. PhD thesis, RWTH Aachen.

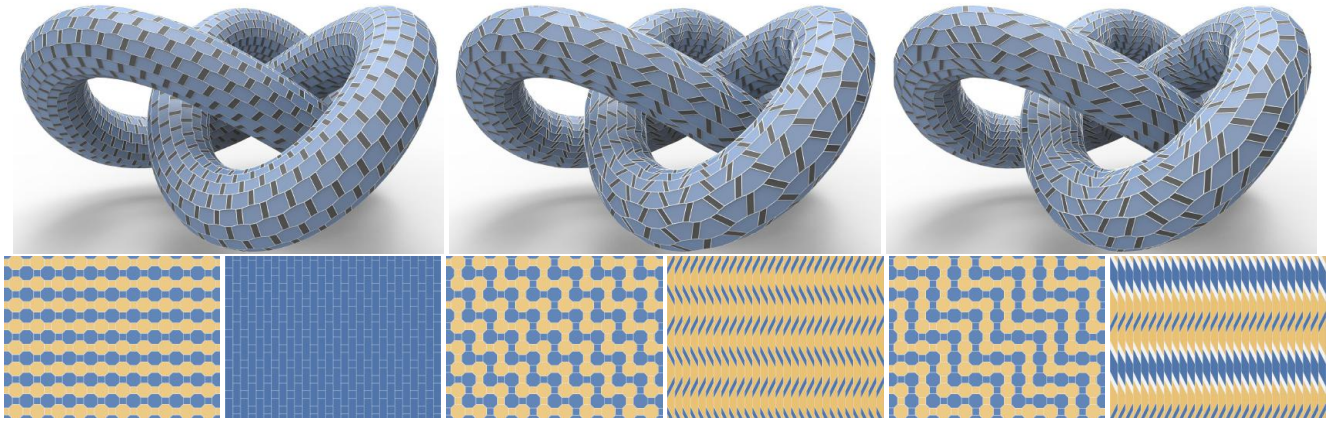


Figure 30: *Three strip decompositions on the knot model for the* $(4, 8, 8)$ *pattern. Figure 1 presents an explanation of the color coding.*

Appendix: conjugacy of patterns

We prove that a primal pattern can be vertically lifted to a paraboloid, S, while every face remains planar, if and only if every primal edge in the pattern is conjugated to and bisected by the corresponding dual edge (i.e., the edge between the conic centers of the neighboring faces). The paraboloid, S, is defined as $z = \kappa_1 x^2 + \kappa_2 y^2$. We currently assume $\kappa_1 \kappa_2 \neq 0$ and refer to this assumption later. We consider the induced quadratic form $\langle \mathbf{a}, \mathbf{b} \rangle := \kappa_1 x_a x_b + \kappa_2 y_a y_b$ on the (x, y)-plane and denote the squared norm $|\mathbf{a}|^2 = \langle \mathbf{a}, \mathbf{a} \rangle$ accordingly. The conjugacy relation is thus $\langle \mathbf{a}, \mathbf{b} \rangle = 0$. We prove our claim using the following:

Lemma 1. *A polygon with vertices* $(\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n)$ *on the* x, y*-plane, inscribed to a conic of the form* $|\mathbf{p} - \mathbf{c}|^2 = \gamma,\ \gamma \in \mathbb{R}$*, remains planar when lifted to* S*.*

Proof. The lifted points of $\mathbf{p}_i$ on the paraboloid are on the intersection of S of the vertical cylinder C extruding the conic: $\kappa_1 (x - x_c)^2 + \kappa_2 (y - y_c)^2 = \gamma$. Thus, they also lie on any linear combination of C and S. The $C - S$ surface is a plane, as the quadratic terms cancel out. □

Lemma 2. *Let* $\mathbf{p}_j - \mathbf{p}_i$ *be a primal edge conjugate to the dual edge* $\mathbf{c}_j - \mathbf{c}_i$ *and bisected by* $\mathbf{c}_j - \mathbf{c}_i$ *at point* $\mathbf{d}$*. Then,* $|\mathbf{p}_i - \mathbf{c}_i|^2 = |\mathbf{p}_j - \mathbf{c}_i|^2$ *and* $|\mathbf{p}_i - \mathbf{c}_j|^2 = |\mathbf{p}_j - \mathbf{c}_j|^2$*.*

Proof. The conjugacy of $\mathbf{p}_j - \mathbf{p}_i$ and $\mathbf{c}_j - \mathbf{c}_i$ implies $\langle \mathbf{p}_j - \mathbf{p}_i, \mathbf{c}_j - \mathbf{c}_i \rangle = 0$, and the bisecting condition implies $|\mathbf{p}_i - \mathbf{d}|^2 = |\mathbf{p}_j - \mathbf{d}|^2$. Thus:

$$\begin{aligned} |\mathbf{p}_i - \mathbf{c}_i|^2 &= |\mathbf{d} - \mathbf{c}_i + \mathbf{p}_i - \mathbf{d}|^2 \\ &= \langle \mathbf{d} - \mathbf{c}_i, \mathbf{d} - \mathbf{c}_i \rangle + \langle \mathbf{p}_j - \mathbf{d}, \mathbf{p}_j - \mathbf{d} \rangle \\ &= \langle \mathbf{p}_j - \mathbf{c}_i, \mathbf{p}_j - \mathbf{c}_i \rangle. \end{aligned}$$

Similarly, $\langle \mathbf{p}_i - \mathbf{c}_j, \mathbf{p}_i - \mathbf{c}_j \rangle = \langle \mathbf{p}_j - \mathbf{c}_j, \mathbf{p}_j - \mathbf{c}_j \rangle$. □

Corollary 1. ***Consistency to planar lifting:*** *A primal pattern on the* (x, y)*-plane with every edge conjugate to and bisected by the dual edge is liftable to* S *while the faces remain planar.*

Proof. Due to Lemma 2, for a given polygon $\{\mathbf{p}_i\}$ and center $\mathbf{c}$, we have $\langle \mathbf{p}_i - \mathbf{c}, \mathbf{p}_i - \mathbf{c} \rangle = \gamma$ for some constant $\gamma \in \mathbb{R}$, and thus the vertices are on a planar conic and can be lifted to S while keeping face planarity, due to Lemma 1. □

We next prove how planar lifting leads to consistency.

Lemma 3. *The vertical projection of the intersection curve between* S *and a plane* $P := z = ax + by + e$ *onto the* (x, y)*-plane is a conic of the form* $|\mathbf{p} - \mathbf{c}|^2 = \gamma,\ \gamma \in \mathbb{R}$*.*

Proof. The surface obtained by the $P - S$ subtraction is a vertical cylinder passing through the intersection curve, which intersects the (x, y)-plane with a conic of the form $\kappa_1 (x_p - x_c)^2 + \kappa_2 (y_p - y_c)^2 = |\mathbf{p} - \mathbf{c}|^2 = \gamma$, where $x_c = \frac{a}{2\kappa_1}$, $y_c = \frac{b}{2\kappa_2}$, and $\gamma \in \mathbb{R}$. □

Lemma 4. *On the* (x, y)*-plane, if two similar conics* $c_i : |\mathbf{p} - \mathbf{c}_i|^2 = \gamma_i$ *and* $c_j : |\mathbf{p} - \mathbf{c}_j|^2 = \gamma_j$ *intersect at* $\mathbf{p}_i$ *and* $\mathbf{p}_j$*, then* $\mathbf{p}_j - \mathbf{p}_i$ *is conjugate to and bisected by* $\mathbf{c}_j - \mathbf{c}_i$*.*

Proof. The subtraction of the two conics, $c_j - c_i$, produces a line l defined by $\langle \mathbf{p}, \mathbf{c}_j - \mathbf{c}_i \rangle = const.$ As both $\mathbf{p}_i$ and $\mathbf{p}_j$ are on l, $\langle \mathbf{p}_j - \mathbf{p}_i, \mathbf{c}_j - \mathbf{c}_i \rangle = 0$.

To show that bisection holds as well, we apply a shearing transformation so that $\bar{\mathbf{c}}_j - \bar{\mathbf{c}}_i$ and $\bar{\mathbf{p}}_j - \bar{\mathbf{p}}_i$ become orthogonal and thus aligned with the axes. As $\bar{c}_i$ and $\bar{c}_j$ are both reflectively symmetric with respect to $\bar{\mathbf{c}}_j - \bar{\mathbf{c}}_i$, so are their intersections. The midpoint of $\mathbf{p}_j - \mathbf{p}_i$ therefore lies on $\mathbf{c}_j - \mathbf{c}_i$ before the shearing transformation. □

Corollary 2. ***Planar lifting to consistency:*** *Consider a pattern on the* (x, y)*-plane which is the vertical projection of a polyhedral pattern inscribed on* S*. Then, there exists a dual pattern with each edge conjugate to and bisecting the corresponding primal edge.*

Proof. Due to Lemma 3, the faces of the primal pattern are all inscribed to conics with the form $|\mathbf{p} - \mathbf{c}_i|^2 = \gamma_i,\ \gamma_i \in \mathbb{R}$. Consider the dual pattern formed by connecting the adjacent face circumconic centers. Then, the primal edges $\mathbf{p}_j - \mathbf{p}_i$ are conjugate to and bisected by the dual edges $\mathbf{c}_j - \mathbf{c}_i$ due to Lemma 4. □

Cylindrical paraboloids When $\kappa_1 \kappa_2 = 0, \kappa_1 \neq \kappa_2$, Corollary 1 still holds, but Corollary 2 is no longer generally exact, as the proof of Lemma 3 requires $\kappa_1 \neq 0$ and $\kappa_2 \neq 0$. The reason for this is that the conics made by this condition are pairs of straight lines without any center, and thus the duality is not well defined. However, by identifying centers, this is still true in the limit of our deformation scheme, as we continuously deform regular tilings from elliptic to hyperbolic conics with well-defined centers. Thus, we have a well-defined dual center obeying these conditions.