

General Planar Quadrilateral Mesh Design Using Conjugate Direction Field

Yang Liu* Weiwei Xu* Jun Wang† Lifeng Zhu‡ Baining Guo* Falai Chen† Guoping Wang‡

*Microsoft Research Asia †University of Science and Technology of China ‡Peking University

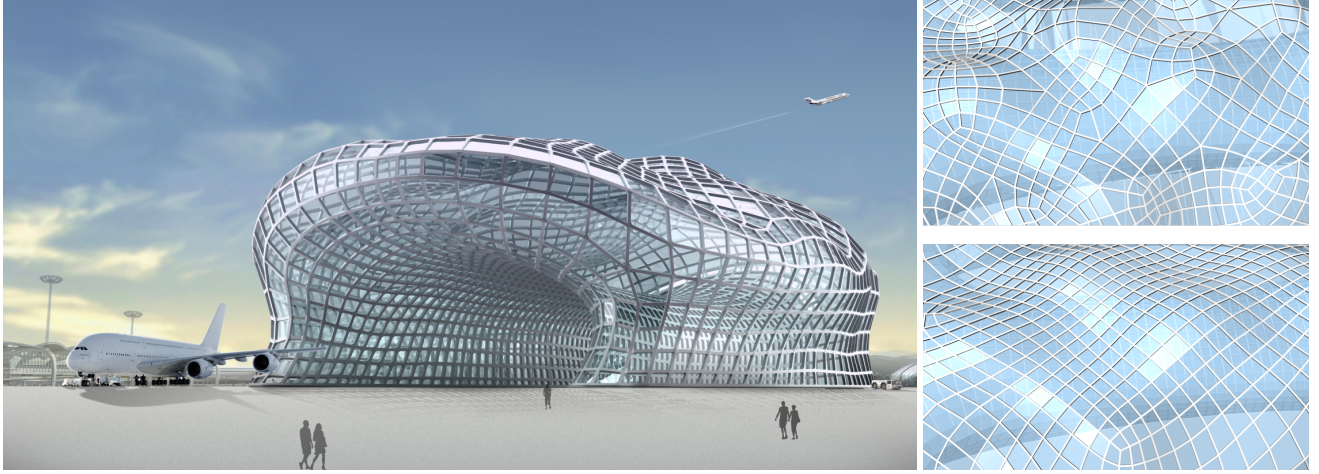


Figure 1: Left: An airport terminal model with planar quad faces generated by our conjugate direction field method. The maximum value of the planarity measure (the angular difference in degrees between the sum of four internal angles of a quad face and 360°) is 0.05° . Right: A comparison of the planar quad mesh on the roof of this model from the principal curvature network (top) and our method (bottom). Our method allows us to control the layout of the planar quad mesh and reduces the number of singularities (non-four-valence vertices).

Abstract

We present a novel method to approximate a freeform shape with a planar quadrilateral (PQ) mesh for modeling architectural glass structures. Our method is based on the study of conjugate direction fields (CDF) which allow the presence of $\pm k/4 (k \in \mathbb{Z})$ singularities. Starting with a triangle discretization of a freeform shape, we first compute an as smooth as possible conjugate direction field satisfying the user’s directional and angular constraints, then apply mixed-integer quadrangulation and planarization techniques to generate a PQ mesh which approximates the input shape faithfully. We demonstrate that our method is effective and robust on various 3D models.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems;

Keywords: planar quadrilateral mesh, conjugate direction field, architectural geometry

Links: [DL](#) [PDF](#)

1 Introduction

Planar quadrilateral (PQ) meshes are essential in architectural geometry for discretizing a freeform architectural structure with planar quad faces [Glymph et al. 2004; Liu et al. 2006; Pottmann et al. 2007b], and the study of PQ meshes is now an important topic of discrete differential geometry [Pottmann and Wallner 2008; Bobenko and Suris 2008]. Its continuous counterpart, in differential geometry, is the conjugate curve network [Sauer 1970; Liu et al. 2006; Bobenko and Suris 2008], which is defined to be two families of one-parameter curves that cover a smooth surface, and their tangent vectors \mathbf{v}, \mathbf{w} at an arbitrary point \mathbf{x} on a surface are conjugate (see its formal definition in Section 3). These two families of tangent directions \mathbf{v}, \mathbf{w} form a general cross field without the requirement of orthogonality, which we call a *conjugate direction field* (CDF) hereafter. The layout of a PQ mesh can be controlled through the design of the CDF.

It has been recognized that an intuitive design tool for smooth CDFs is desirable for architects to control the layout of the PQ mesh [Pottmann et al. 2007a; Eigensatz et al. 2010]. Unfortunately, there is no general solution currently available for CDF design. Existing techniques can handle two special cases. Principal directions, as a typical example of CDFs, have been used in [Liu et al. 2006] to produce PQ meshes. Since the principal directions are unique, there are no degrees of freedom left for the architects. A recent representation-vector based CDF design technique in [Zdravce et al. 2010] is capable of producing a smooth CDF via measuring the smoothness of the representation vector field. However, only singularities with indices of $\pm k/2 (k \in \mathbb{Z})$ can be modeled and it fails in handling $\pm k/4 (k \in \mathbb{Z})$ singularities, such as a surface with convex corners (e.g., a round cube).

The main challenge of general CDF design is how to define a correct smoothness measure for a CDF so that singularities of $\pm k/4 (k \in \mathbb{Z})$ are allowed. Since a CDF on two adjacent faces

consists of two pairs of directional vectors, its smoothness can only be measured after the vector association issue is resolved. That is, we need to figure out which vector is associated with which vector between the neighboring conjugate directions. The existing approach in [Zdravec et al. 2010] implies an order in two conjugate directions and thus prohibits the arbitrary association of vectors. Furthermore, note that a CDF is not a rotational symmetry (RoSy) field since the angle between any pair of conjugate directions varies across the surface. The vector association techniques for the RoSy field, such as the period jump technique in [Ray et al. 2008] and the trivial connection in [Crane et al. 2010], cannot be directly applied.

The main contribution of this paper is to propose a general CDF design scheme that enables the user to fully explore the degrees of freedom in a CDF. The key observation is that a CDF is exactly smooth only when the vector association between neighboring conjugate directions can be modeled by a signed-permutation operation. Since the membership of direction vectors in a CDF are not differentiated in this operation, arbitrary types of vector associations can be modeled to allow the appearance of $\pm k/4$ singularities. We show that the signed-permutation condition for a smooth CDF can be converted into a proper smoothness measure which can be computed explicitly. Similar to the RoSy field smoothing objective function in [Ray et al. 2009], our smoothness measure is only a summation of trigonometric functions. This significantly facilitates the direction field optimization to seek an as smooth as possible CDF on the surface. A side benefit of our measure is that it allows the direct control of the angle between conjugate directions to avoid *self-conjugate* directions (a direction that is conjugate to itself).

After the design of CDF, we adapt the global parametrization technique in [Bommes et al. 2009] to trace the iso-lines following the conjugate directions, and then extract an initial quad mesh through the intersections of the iso-lines. A perturbation algorithm is then applied to optimize the quad mesh into a PQ mesh. Figure 1 illustrates an example from our method.

We have evaluated our method on a variety of models, including architectural models with highly-varied curvature distributions and 3D freeform models, such as the Stanford Bunny. Experimental results demonstrate the effectiveness and robustness of our method in generating high-quality PQ meshes.

2 Related Work

***N*-RoSy Field** represents N coupled directions which are invariant under rotations of an integer multiple of $\frac{2\pi}{N}$. Therefore, the N -RoSy field design algorithm should be able to handle the vector association issue to model fractional singularities. Hertzmann and Zorin [2000] adopted an angle formulation to formulate the smoothness energy of a cross field. Vector association is achieved using integer variables, which are eliminated through the trigonometric function in the nonlinear optimization procedure.

Ray et al. [2008] proposed a period-jump based discretization of a 4-RoSy field on a surface, where the period-jump is an integer variable encoded at an edge for the vector association. Their method built a linear system to solve for a globally smooth 4-RoSy field. However, the direct rounding scheme in their method might lead to undesirable singularities in the resulting field. To solve this problem, a geometry-aware method [Ray et al. 2009] was developed to control the topology of an N -RoSy field by integrating the filtered defect angles into a smoothness energy function. Bommes et al. [2009] proposed a mixed-integer solver for the design of an N -RoSy field. Instead of direct rounding, their method iteratively rounds the integer variables to further reduce the smoothness energy and the number of singularities. In contrast, Palacios and Zhang [2007] used representation vectors to control the singularity

of the N -RoSy field. They also provided an intuitive interface for design and editing. Recently, an elegant method based on the trivial connection [Crane et al. 2010] simplified the design of a smooth N -RoSy field by solving a linear system only.

Our algorithm is inspired by the N -RoSy field design algorithm. However, we adopt the signed-permutation technique to handle the varying angles between the conjugate directions.

Quadrangulation is to compute a quadrilateral structure on a surface and it is well studied in the mesh generation community. With an augmented vector/cross field on a surface, curve tracing or global parameterization methods are developed to generate a quad mesh [Alliez et al. 2003; Boier-Martin et al. 2004; Tong et al. 2006; Ray et al. 2006; Kälberer et al. 2007; Bommes et al. 2009]. The Morse-Smale complex of a scalar function, as another approach, can generate a global quadrilateral structure and automatically optimize the distribution of singularities [Dong et al. 2006; Huang et al. 2008]. In this paper, we adapt the mixed integer quadrangulation method [Bommes et al. 2009] to guarantee the global topological structure and produce all-quad meshes.

PQ mesh is preferable for the purpose of fabrication in architecture [Glymph et al. 2004], especially for glass structures. Liu et al. [2006] extracted quad meshes from the principal curvature lines and developed a PQ perturbation algorithm to enforce planarity of quad faces. Recently Zdravec et al. [2010] cast the design of the conjugate curve network into a vector field design problem, and the conjugate directions are computed after the optimization of a vector field. However, as mentioned in Section 1, their algorithm has a limitation in modeling singularities of the index $\pm k/4$ ($k \in \mathbb{Z}$). In addition, the angle between conjugate directions cannot be controlled directly. In comparison, our method completely solves the vector association issue in a CDF so that singularities of $\pm k/4$ can be well handled. Our algorithm also enforces the explicit constraint on the angle between two conjugate directions to avoid the appearance of self-conjugacy.

3 CDF On Triangle Meshes

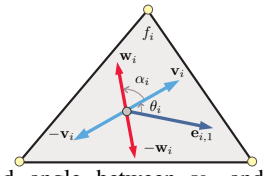
It is well known that, on a smooth surface $S \subset \mathbb{R}^3$, two tangent vectors $\mathbf{v}_p, \mathbf{w}_p$ in the tangential space $T_p(S)$ at the point $p \in S$ are conjugate if and only if the bilinear form $\Pi_p(\mathbf{v}_p, \mathbf{w}_p) = 0$ [Do Carmo 1976], where Π_p is the second fundamental form at p . If \mathbf{v}_p and \mathbf{w}_p are treated as two vectors in \mathbb{R}^3 , the preceding equality can be written in the following form:

$$\kappa_{p,1}(\mathbf{v}_p \cdot \mathbf{e}_{p,1})(\mathbf{w}_p \cdot \mathbf{e}_{p,1}) + \kappa_{p,2}(\mathbf{v}_p \cdot \mathbf{e}_{p,2})(\mathbf{w}_p \cdot \mathbf{e}_{p,2}) = 0 \quad (1)$$

where $\kappa_{p,1}, \kappa_{p,2}$ are the principal curvatures at p , and $\mathbf{e}_{p,1}, \mathbf{e}_{p,2}$ are the corresponding principal directions.

Following the direction field discretization method in [Ray et al. 2009], we define a CDF on a triangular mesh to be two families of tangent direction fields $\{\mathbf{v}, \mathbf{w}\}$ sampled at each triangle f , and they are conjugate to each other, i.e., Eqn. 1 holds at each triangle f . In the following, we first introduce the notations for a CDF, and then describe how to define its signed-permutation-based smoothness and singularity index, which are critical to the design of a CDF.

As shown in the right inset, a CDF on a triangle f_i is four vectors $\{\mathbf{v}_i, \mathbf{w}_i, -\mathbf{v}_i, -\mathbf{w}_i\}$, and they can be parameterized by two scalar parameters $\{\theta_i, \alpha_i\}$, where θ_i is the oriented angle between $\mathbf{e}_{i,1}$ and \mathbf{v}_i , and α_i is the oriented angle between \mathbf{v}_i and \mathbf{w}_i . Therefore, we have $\mathbf{v}_i = \langle \cos \theta_i, \sin \theta_i \rangle^T$ and $\mathbf{w}_i = \langle \cos(\theta_i + \alpha_i), \sin(\theta_i + \alpha_i) \rangle^T$.



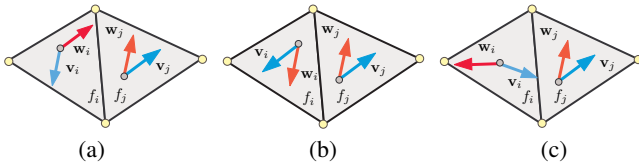


Figure 2: (a-b) Two cases of vector associations in a smooth CDF and the corresponding signed-permutation matrices: $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$ and $\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$. (c) An example of a pair of non-smooth conjugate directions.

3.1 The Smoothness of a CDF

Similar to the N -RoSy field [Ray et al. 2008], the smoothness of a CDF is also computed at each edge e_{ij} incident to two triangles f_i and f_j . In fact, we compute the angle difference between the associated direction vectors, which is formally called a *discrete connection* in [Crane et al. 2010], at each edge e_{ij} to measure the change of the conjugate directions. Due to the fact that there are two directions on a face and the angle between them varies across the mesh, we treat a CDF as two coupled 2-RoSy fields on the mesh to measure its smoothness. Consequently, two angle differences $C_1(e_{ij})$ and $C_2(e_{ij})$ need to be computed at edge e_{ij} :

$$\begin{aligned} C_1(e_{ij}) &= (\theta_j + q\alpha_j) + r_{ij} - \theta_i + p_1\pi; \\ C_2(e_{ij}) &= (\theta_j + (1-q)\alpha_j) + r_{ij} - (\theta_i + \alpha_i) + p_2\pi \end{aligned} \quad (2)$$

where r_{ij} is the rotation angle between two local reference frames $\mathbf{e}_{i,1}$ on f_i and $\mathbf{e}_{j,1}$ on f_j . $q \in \{0, 1\}$ is used to choose \mathbf{v}_j or \mathbf{w}_j at f_j for associating vectors, and p_1 and p_2 are integers serving as the period jumps in the N -RoSy field design [Ray et al. 2008]. Note that q plays an important role here in modeling the associations of vectors.

Smoothness Measure. To produce a smooth CDF, an ideal algorithm needs to minimize the magnitudes of $C_1(e_{ij})$ and $C_2(e_{ij})$ simultaneously. However, note that this formulation builds a nonlinear relationship between the $(0, 1)$ -integer variable q and the floating-point variable α_j , so it dramatically increases the complexity for further optimization. As a result, it is very difficult to design an efficient algorithm to minimize this nonlinear mixed-integer optimization problem. Neither the mixed-integer technique employed in [Bommes et al. 2009] nor the trivial connection technique in [Crane et al. 2010] can be used to solve this problem.

However, note that when both $C_1(e_{ij}), C_2(e_{ij})$ vanish, i.e., the CDF is perfectly smooth, the corresponding directions on two adjacent faces can be permuted to each other, i.e., we have

$$(\mathbf{v}_i | \mathbf{w}_i) \mathbf{P}_{ij} = (\tilde{\mathbf{v}}_j | \tilde{\mathbf{w}}_j).$$

Here \mathbf{P}_{ij} is a 2×2 signed-permutation matrix, i.e., \mathbf{P}_{ij} is a $0, 1, -1$ -matrix with one nonzero entry in each row and each column. $(\tilde{\mathbf{v}}_j, \tilde{\mathbf{w}}_j)$ are the representations of $(\mathbf{v}_j, \mathbf{w}_j)$ in the local reference frames at f_i by using a hinge map as a local isometric parametrization, i.e., $\tilde{\mathbf{v}}_j = \langle \cos(\theta_j + r_{ij}), \sin(\theta_j + r_{ij}) \rangle^T$ and $\tilde{\mathbf{w}}_j = \langle \cos(\theta_j + \alpha_j + r_{ij}), \sin(\theta_j + \alpha_j + r_{ij}) \rangle^T$. Figure 2a and 2b illustrate two cases of vector association in a smooth CDF and their corresponding \mathbf{P}_{ij} s.

From Eqn. 3.1, \mathbf{P}_{ij} can be estimated from the CDF using the following formula:

$$\mathbf{P}_{ij} = (\mathbf{v}_i | \mathbf{w}_i)^{-1} (\tilde{\mathbf{v}}_j | \tilde{\mathbf{w}}_j) = \frac{1}{\sin \alpha_i} \mathbf{C}_{ij}$$

where $\mathbf{C}_{ij} = \begin{pmatrix} \sin(\theta_i - \theta_j + \alpha_i - r_{ij}) & \sin(\theta_i - \theta_j + \alpha_i - \alpha_j - r_{ij}) \\ \sin(\theta_j - \theta_i + r_{ij}) & \sin(\theta_j - \theta_i + \alpha_j + r_{ij}) \end{pmatrix}$. When the magnitude of $C_1(e_{ij})$ or $C_2(e_{ij})$ is large, the computed \mathbf{P}_{ij} from the above equations can be far from a signed-permutation matrix. Figure 2c illustrates an example of a non-smooth CDF on f_i and f_j .

Motivated by the above observation, we resort to a signed-permutation matrix approach to define the smoothness of a CDF directly. The smoothness measure of a CDF is then based on the computation of the deviation from \mathbf{P}_{ij} to the signed permutation matrix group. It can be described by the following proposition:

Proposition 1 \mathbf{P}_{ij} is a signed-permutation matrix iff the following two conditions hold: $\mathbf{P}_{ij}^{-1} = \mathbf{P}_{ij}^T$ and $\frac{C_1(e_{ij}) + C_2(e_{ij})}{2} = 0$.

Proof. The first condition is due to the fact that any signed permutation matrix is an orthogonal matrix. It can be re-organized into:

$$\mathbf{D}_{ij} := (\mathbf{v}_i | \mathbf{w}_i) (\mathbf{v}_i | \mathbf{w}_i)^T - (\tilde{\mathbf{v}}_j | \tilde{\mathbf{w}}_j) (\tilde{\mathbf{v}}_j | \tilde{\mathbf{w}}_j)^T = \mathbf{0}.$$

The second condition comes from the fact that both $C_1(e_{ij})$ and $C_2(e_{ij})$ vanish when \mathbf{P}_{ij} is a signed-permutation matrix. By substituting Eqn. 2 into this condition and multiplying it by 4, we have $4(\theta_i + \frac{\alpha_i}{2}) = 4(\theta_j + \frac{\alpha_j}{2}) + 4r_{ij} + 2(p_1 + p_2)\pi$. To eliminate the two integers p_1 and p_2 , we take the cosine and sine on both sides and get the following formula:

$$\mathbf{E}_{ij} := \begin{pmatrix} \cos(4\theta_i + 2\alpha_i) - \cos(4\theta_j + 2\alpha_j + 4r_{ij}) \\ \sin(4\theta_i + 2\alpha_i) - \sin(4\theta_j + 2\alpha_j + 4r_{ij}) \end{pmatrix} = \mathbf{0}.$$

From the above derivation, it is obvious that $\mathbf{D}_{ij} = \mathbf{0}$ and $\mathbf{E}_{ij} = \mathbf{0}$ are necessary conditions when \mathbf{P}_{ij} is a signed-permutation matrix. We only need to prove that they are also sufficient conditions. From $\mathbf{E}_{ij} = \mathbf{0}$, we have:

$$\theta_j + r_{ij} = \theta_i + \frac{\alpha_i - \alpha_j}{2} + \frac{k\pi}{2}, k \in \mathbb{Z}. \quad (3)$$

By substituting Eqn. 3 into $\mathbf{D}_{ij} = \mathbf{0}$, we can derive that $\cos \alpha_i = (-1)^k \cos \alpha_j$. Therefore, if k is even, then $\alpha_j = \alpha_i + 2l\pi, l \in \mathbb{Z}$. We have:

$$\begin{cases} \theta_j + r_{ij} = \theta_i + (\frac{k}{2} - l)\pi \\ \theta_j + \alpha_j + r_{ij} = \theta_i + \alpha_i + (\frac{k}{2} + l)\pi \end{cases}, l \in \mathbb{Z}.$$

If k is odd, then $\alpha_j = (2l + 1)\pi - \alpha_i, l \in \mathbb{Z}$, and we have:

$$\begin{cases} \theta_j + r_{ij} = \theta_i + \alpha_i + (\frac{k-1}{2} - l)\pi \\ \theta_j + \alpha_j + r_{ij} = \theta_i + (\frac{k+1}{2} + l)\pi \end{cases}, l \in \mathbb{Z}.$$

It is easy to verify that $\{\mathbf{v}_i, \mathbf{w}_i\}$ can be signed-permuted to $\{\tilde{\mathbf{v}}_j, \tilde{\mathbf{w}}_j\}$. Therefore \mathbf{P}_{ij} is a signed permutation matrix. ■

Having got the equivalent conditions for \mathbf{P}_{ij} to be a signed permutation matrix, we define the smoothness of a CDF S_{ij} at edge e_{ij} as follows:

$$\begin{aligned} S_{ij} := & \|\mathbf{D}_{ij}\|_F^2 + \|\mathbf{E}_{ij}\|_2^2 = 4 + \cos(2\alpha_i) + \cos(2\alpha_j) - \\ & \cos(2(\theta_i + \alpha_i - \theta_j - r_{ij})) - \cos(2(\theta_j + \alpha_j - \theta_i + r_{ij})) - \\ & \cos(2(\theta_j - \theta_i + r_{ij})) - \cos(2(\theta_j + \alpha_j + r_{ij} - \theta_i - \alpha_i)) - \\ & 2 \cos(4(\theta_i + \frac{\alpha_i}{2} - \theta_j - \frac{\alpha_j}{2} - r_{ij})), \end{aligned} \quad (4)$$

where $\|\cdot\|_F$ is the Frobenius norm.

Remark: If α_i and α_j are $\frac{\pi}{2}$, we have $S_{ij} = \|\mathbf{E}_{ij}\|_2^2 = 2 - 2 \cos(4\theta_1 - 4\theta_2 - 4r_{ij})$ which is the smoothness measure of a cross field used in [Hertzmann and Zorin 2000; Ray et al. 2009].

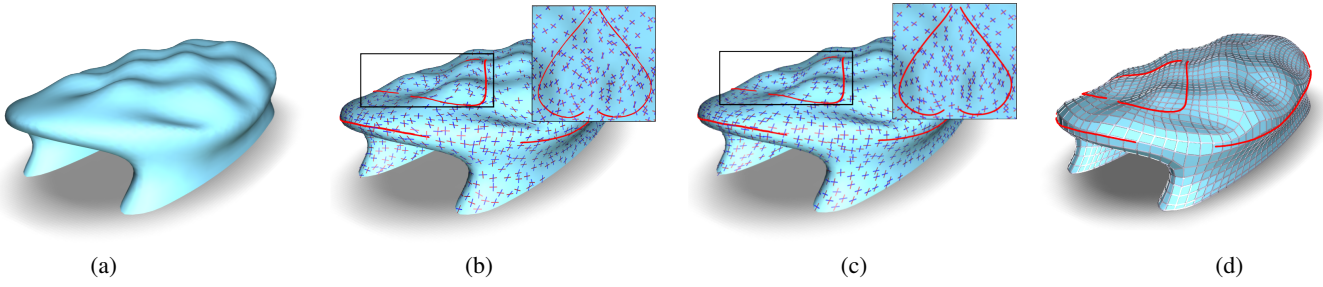


Figure 3: CDF design on an airport terminal model. (a) The original model. (b) An initial CDF from user-specified strokes (red lines). (c) The optimized CDF. (d) The resulting PQ mesh.

3.2 Index of Singularity

The index of singularity is a fundamental concept introduced by Poincaré and Hopf to identify the singularities of a vector/direction field. For a CDF, we define the index of a CDF at a vertex u as

$$I(u) = \frac{K_d(u) + \sum_{e_{ij} \in N(u)} \frac{C_1(e_{ij}) + C_2(e_{ij})}{2}}{2\pi} \quad (5)$$

where $K_d(u)$ denotes the angle defect of u (i.e., 2π minus the summation of angles adjacent to u), and $N(u)$ denotes the edges incident to u and the average of $C_1(e_{ij})$ and $C_2(e_{ij})$ measures the difference between two pairs of conjugate directions on two neighboring faces at edge e_{ij} .

Actually $\frac{C_1(e_{ij}) + C_2(e_{ij})}{2}$ is exactly the angle difference at e_{ij} of a 4-RoS field \mathcal{B} where its member direction vectors are the bisectors of \mathbf{v}_i and \mathbf{w}_i . Our observation is from the following simple calculation:

$$\frac{C_1(e_{ij}) + C_2(e_{ij})}{2} = \left(\theta_j + \frac{\alpha_j}{2} + r_{ij}\right) - \left(\theta_i + \frac{\alpha_i}{2}\right) + \frac{(p_1 + p_2)\pi}{2}. \quad (6)$$

This fact indicates that the computation of the index of a singularity in a CDF can be performed in its adjoint 4-RoS field \mathcal{B} .

4 CDF Optimization

With the definition of smoothness and index, we are ready to develop a CDF optimization algorithm. Given a triangular mesh and user-specified directional constraints on a subset of faces, our direction field optimization algorithm seeks an as smooth as possible CDF satisfying these constraints. The overall workflow of CDF optimization is illustrated in Figure 3.

In the direction field optimization, we seek for optimal θ_i, α_i at each triangle by minimizing the smoothness energy under the conjugacy constraints, angular constraints, and directional constraints.

Smoothness energy function is used to measure the smoothness of the CDF. It is a summation of S_{ij} defined in Eqn. 4 over each edge:

$$E_s = \sum_{e_{ij}} S_{ij}. \quad (7)$$

Conjugacy constraint. The conjugacy at each face can be measured through Eqn. 1. Since we set the local reference frame to be the principal curvature directions, the conjugacy constraint on f_i is:

$$C_{f_i} = \kappa_{i,1} \cos(\theta_i) \cos(\theta_i + \alpha_i) + \kappa_{i,2} \sin(\theta_i) \sin(\theta_i + \alpha_i) = 0.$$

We adopt the normal cycle technique in [Cohen-Steiner and Morvan 2003] to compute the principal curvatures $\kappa_{i,1}, \kappa_{i,2}$ and principal directions at face f_i . Curvature tensors are first estimated at vertices, and then smoothed to filter out the discretization noise [Alliez et al. 2003]. The curvature tensor at f_i is approximated by averaging the curvature tensors at its vertices.

In our optimization, we introduce an inequality conjugacy constraint at each face:

$$-\epsilon_c c \leq C_i \leq \epsilon_c c \quad (8)$$

where $c = \max_i(|\kappa_{i,1}|, |\kappa_{i,2}|)$, and ϵ_c is a small value (default: 0.001) to control how well the conjugacy condition should be satisfied. Our inequality formulation of the conjugacy constraint avoids numerical instability due to unreliable curvature tensor estimation at noisy areas.

Directional constraint. Control of the local orientation of the CDF is critical in our algorithm. We provide a stroke-based interface for the user to specify the directional constraints on the mesh. We support this constraint by introducing the following inequality constraint:

$$\alpha_d \leq \psi_{i_c} - \theta_{i_c} \leq \alpha_d$$

where i_c denotes the face which contains the directional constraint, ψ_{i_c} is the angle between user-specified directions and local reference vectors on f_{i_c} , and α_d is a user-supplied angle to control the angular difference between the conjugate directions and the user's input (default: 10°).

Angular constraint. Small angles between two conjugate directions need to be avoided to guarantee the quality of the resulting PQ mesh. We thus set the angular constraint at each face to be:

$$\alpha_s \leq \alpha_i \leq \pi - \alpha_s.$$

Here α_s is the minimal angle defined by the user (default: 15°). The angular constraint is not added to faces with directional constraints due to the possible conflict.

Initialization. Since we are dealing with a nonlinear optimization problem, an properly initialized CDF has to be determined to start the optimization. Our initialization procedure is to mimic parallel transport operation to propagate the conjugate directions at constrained faces to the whole mesh. However, since conjugate directions are not unique at each face, we choose to first generate a smooth bisector direction field \mathcal{B} which is 4-RoS, then use the curvature information to compute a pair of conjugate directions on each face. The initialization procedure is as follows:

1. Compute conjugate directions for each face f_{i_c} with a directional constraint by solving Eqn. 1 where \mathbf{v}_i or \mathbf{w}_i is set by this direction. Set the bisector direction at f_{i_c} to

$\mathbf{b}_{ic} = \frac{\mathbf{v}_{ic} + \mathbf{w}_{ic}}{\|\mathbf{v}_{ic} + \mathbf{w}_{ic}\|}$, and denote the oriented angle between \mathbf{b}_{ic} and $\mathbf{e}_{ic,1}$ by ϕ_{ic} . Push these faces with directional constraints into a queue Q and label them *visited*. For other faces without directional constraints, label them *unvisited*.

2. Generate a 4-RoSy field by a FIFO propagation. Pop a face f_i from Q . For each of its neighboring faces f_j , push it into the queue if f_j is *unvisited*. Label f_j *visited*, and set ϕ_j to minimize $|(\phi_i - (\phi_j + r_{ij})) + k\frac{\pi}{2}|$ (k is an integer value). This push/pop procedure iterates until Q is empty.
3. Compute conjugate directions for each unconstrained face f_i . Two parameters, θ_i and α_i , can be found through the solutions of two equations, $\theta_i + \frac{\alpha_i}{2} = \phi_i$ and $C_{f_i} = 0$. Note that the solution may not exist in the negative Gaussian curvature region. In this case, we simply assign $\theta_i = 0, \alpha_i = \frac{\pi}{2}$.

Figure 3 demonstrates an example of the initialization of a CDF.

An alternative way is to use the mixed-integer technique [Bommes et al. 2009] to generate a smooth \mathcal{B} with less singularities. However, our approach is lightweight and fast to compute. It is enough to produce good initial CDFs for further optimization in our experiments.

Optimization. The smooth energy function and the inequality constraints form a nonlinear constrained optimization problem. Since the number of variables and constraints can be very large (proportional to the number of faces), we use an augmented Lagrangian method to solve it efficiently.

The augmented Lagrangian method converts a nonlinear constrained problem $\{\min f(\mathbf{x}), \text{s.t. } c_i(\mathbf{x}) = 0, i = 1, \dots, r; c_i(\mathbf{x}) \geq 0, i = r + 1, \dots, n\}$ to an unconstrained problem:

$$\min \varphi(\mathbf{x}, \lambda, \sigma) = f(\mathbf{x}) - \lambda^T \mathbf{d}(\mathbf{x}) + \frac{1}{2} \sigma \mathbf{d}(\mathbf{x})^T \mathbf{d}(\mathbf{x})$$

where

$$\mathbf{d}_i(\mathbf{x}) = \begin{cases} c_i(\mathbf{x}) & \text{if } i \leq r \text{ or } c_i(\mathbf{x}) \leq \frac{\lambda_i}{\sigma}; \\ \frac{\lambda_i}{\sigma} & \text{otherwise.} \end{cases}$$

λ is the Lagrangian multiplier associated with each constraint and σ is the penalty parameter. They are iteratively updated to tighten the tolerances of the constraint error [Nocedal and Wright 1999]. At each iteration, the limited-memory BFGS (L-BFGS) algorithm [Liu and Nocedal 1989] is adopted to solve the unconstrained optimization problem in our implementation.

Singularity editing. With the index formula defined in Eqn. 5, we are able to detect all the singularities of a CDF via its bisector directional field \mathcal{B} . However, our smoothness function does not penalize the number of singularities directly and the distribution of singularity points may be unsatisfactory. To tackle this, we follow the geometry-aware approach proposed in [Ray et al. 2009] to manipulate singularities of \mathcal{B} , i.e., allow the user to select and edit the singular vertices by *move*, *merge* and *cancel* operations. Recalling that \mathcal{B} is a 4-RoSy field (see Eqn. 6), we denote $B(e_{ij}) = \frac{C_1(e_{ij}) + C_2(e_{ij})}{2}$ as the angle difference of \mathcal{B} defined on edge e_{ij} . Now we sketch the editing steps as follows; please refer to [Ray et al. 2009] for more details:

1. Apply a Gaussian filter to obtain the smoothed angle defects K^{corr} on vertices.
2. Modify K^{corr} according to the editing of singularities: move a singularity of index I from a vertex u_1 to a vertex u_2 by adding $2\pi I$ to $K^{corr}(u_1)$ and subtracting $2\pi I$ from $K^{corr}(u_2)$. The merging and canceling operations can be achieved by the moving operation.

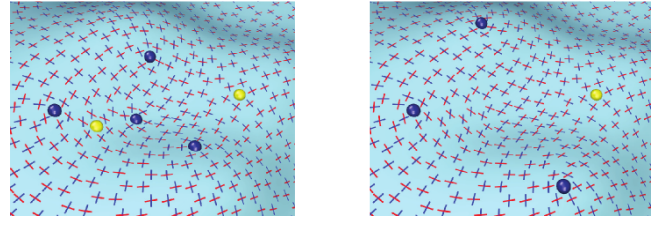


Figure 4: Singularity editing. Left: a CDF with 6 singularities. The blue ball indicates a $1/4$ singularity, and the yellow ball indicates a $-1/4$ singularity. Right: the editing result. Two singularities in the central region are canceled, and the singularity at the top-left is moved away from others.

3. Recompute $B(e_{ij})$ by minimizing $\sum_{e_{ij}} B(e_{ij})^2$ under the constraints $\sum_{e_{ij} \in N(u)} B(e_{ij}) = K^{corr}(u) - K_d(u)$ defined at each vertex u .
4. Modify the rotation angle r_{ij} on each edge by letting r_{ij} be $r_{ij}^0 - B(e_{ij})$, and then optimize the smoothness of CDF. Here r_{ij}^0 is the unmodified rotation angle.

Figure 4 shows an editing result.

5 PQ Mesh Generation

The goal of PQ mesh generation is to find a PQ mesh following the optimized CDF. We first adapt the global parametrization method in [Dong et al. 2006; Bommes et al. 2009] to generate an initial quad mesh, and then improve its quality by planarization.

5.1 Global Parametrization

In the global parametrization, we assign an (s, t) parameter value to each vertex of the input mesh so that its iso-parameter lines on the surface are locally oriented according to the optimized CDF. Specifically, we minimize the following energy function to seek for the optimal (s, t) parameter values:

$$E_p = \sum_{f_i} \text{area}(f_i) \left[\left(\frac{\nabla s_i}{\|\nabla s_i\|} \cdot \mathbf{v}_i^T \right)^2 + \left(\frac{\nabla t_i}{\|\nabla t_i\|} \cdot \mathbf{w}_i^T \right)^2 \right] \quad (9)$$

where $(\mathbf{v}_i^T, \mathbf{w}_i^T) = \text{rot}_{90}(\mathbf{v}_i, \mathbf{w}_i)$ (rot_{90} means rotation counter-clockwise by 90 degrees). Note that we introduce a normalization operator into the objective function to convert the gradient field of (s, t) into a direction field, which is different from the parametrization energy function in the mixed integer technique [Bommes et al. 2009]. This means that our formulation focuses on the orientation alignment and does not care about the length mismatch. It can lead to better alignment between the parameter lines and the optimized CDF. The advantage of the formulation in Eqn. 9 is illustrated in Fig. 5.

However, Eqn. 9 is a nonlinear energy function with many integer variables introduced at the topology cut and singularities [Bommes et al. 2009]. We thus design a nonlinear mixed integer solver to solve it. The solver performs three steps to minimize the energy function until convergence: (a) optimize the function with the L-BFGS method; (b) round an integer variable to its nearest integer and set this variable as a constant; goto (a) until all the integer variables are fixed; (c) optimize the function until the L-BFGS method converges.

The number of integer variables can significantly influence the speed of the solver. We observe that the topology cut can be seg-

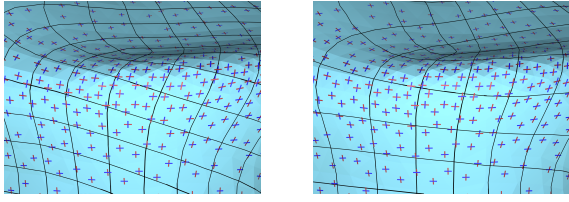


Figure 5: A zoom-in view of the airport model (parameterized lines are in black). Left: the mixed technique approach; right: our approach. The parameter lines generated by our approach aligns to the CDF better due to nonlinear optimization.

mented into polylines whose end points are located at the singularities, the branching points of the cut or some boundary vertices. If the edges of each polyline share the same type of signed permutation, their integer translational variables should be the same. Utilizing these observations, we can introduce only one pair of integer translational variables on each segment to reduce the number of integer variables dramatically. For instance, the number of integer variables in the optimization of the airport model (Fig. 3) is reduced from 366 to 64.

5.2 Planarity Optimization

Similar to the PQ perturbation algorithm in [Liu et al. 2006; Zdravcevic et al. 2010], our planarity optimization is formulated as a nonlinear constrained optimization problem and solved by the augmented Lagrangian method.

$$\begin{aligned} E_f &= w_{fair} E_{fair} + w_{2nd} E_{2nd} + w_{dist} E_{dist} \\ \text{subject to: } &\phi_{ij}^1 + \phi_{ij}^2 + \phi_{ij}^3 + \phi_{ij}^4 = 2\pi; \end{aligned} \quad (10)$$

where the objective function E_f includes two fairness terms, E_{fair} and E_{2nd} , and a distance term E_{dist} to keep the optimized mesh close to the original mesh. They can be represented by:

$$\begin{aligned} E_{fair} &= \sum_{i,j} (\|\delta_{i,j,1}\|^2 + \|\delta_{i,j,2}\|^2) \\ E_{2nd} &= \sum_{i,j} (\|\delta_{i,j,1} - \delta_{i,j,1}^0\|^2 + \|\delta_{i,j,2} - \delta_{i,j,2}^0\|^2) \\ E_{dist} &= \sum_{i,j} \|v_{i,j} - v_{i,j}^0\|^2 \end{aligned}$$

where $v_{i,j}$ and $v_{i,j}^0$ denote the position of the optimized vertex and the original vertex on the quad mesh. For the non-singular vertex $v_{i,j}$, $\delta_{i,j,1} = v_{i+1,j} + v_{i-1,j} - 2v_{i,j}$, $\delta_{i,j,2} = v_{i,j+1} + v_{i,j-1} - 2v_{i,j}$, and $\delta_{i,j,1}^0, \delta_{i,j,2}^0$ are the original values of $\delta_{i,j,1}$ and $\delta_{i,j,2}$ before optimization. For the singular vertex, δ is the Laplacian operation defined by its one ring neighborhood. The equality constraint ensures the sum of four internal angles, $\phi_{ij}^1, \dots, \phi_{ij}^4$, to be 2π for each quad Q_{ij} in a mesh. This constraint enforces the optimized Q_{ij} to be planar and convex.

6 Experimental Results and Comparisons

We demonstrate the effectiveness of our CDF optimization algorithm through a variety of architectural models and complex 3D models like the Stanford Bunny (see Figure 6 for models and the user’s strokes), and the statistics and timings are reported in Table 1. The quality of planarity is measured by two criterions: (a) δ is the maximum angle difference in degrees between the sum of four internal angles of a quad and 2π . δ is irrelevant to the size of quads and is zero when all the quads are planar and convex. (b) η is the maximum distance between diagonals of the quad face divided by the mean length of diagonals of the quad mesh. η reflects how much the glass panel should be bent and is more relevant to architectural applications. The approximation quality ζ of the generated

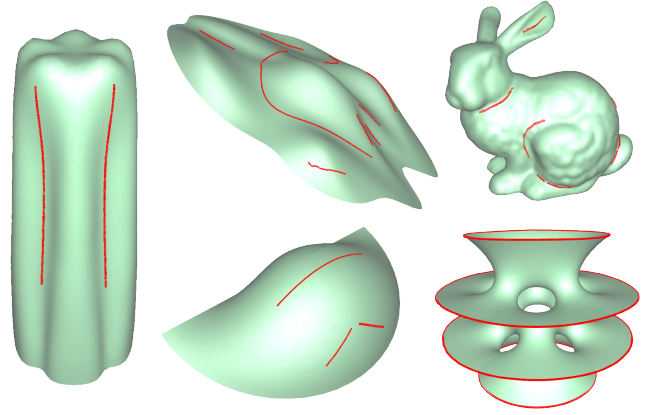


Figure 6: Models with the user-specified strokes overlaid in red. Left: a tower. Top middle: a roof. Bottom middle: a snail-shell. Top right: the Stanford Bunny. Bottom right: a Costa surface.

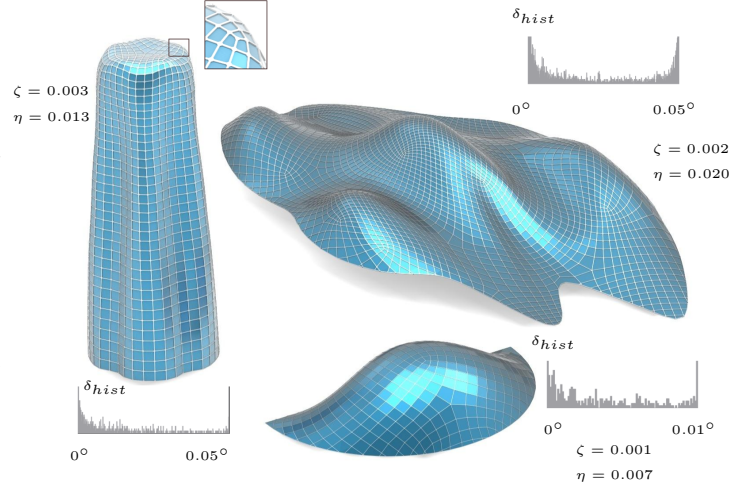


Figure 7: PQ meshes of architectural models. Strokes are shown in Figure 6. The top of the tower model is zoomed to show singularity nodes of valence 3. The histogram measures the distribution of face planarity.

PQ mesh to the original shape is measured by the average distance from vertices on the PQ mesh to the original model normalized by the diagonal length of the bounding box of the model.

Our algorithm can efficiently generate a smooth CDF for models with a highly varied distribution of curvatures. Figure 3 illustrates a design result of an airport terminal model. Its roof is a wavy surface with transition areas from the positive Gaussian curvature to the negative Gaussian curvature. With the user-specified strokes (shown in Figure 3b), our algorithm successfully generates a smooth CDF, and the resulting PQ mesh is shown in Figure 3d. A comparison of PQ meshes on the roof between the principal direction field and our CDF is illustrated in Figure 1. Due to the complex distribution of curvatures, the principal direction field leads to an uneven distribution of quads and more singularities (111 in total) than our method (24 in total).

Figure 7 demonstrates more CDF design results for architectural models. For these models, only simple strokes (shown in Figure 6) are required to guide the generation of smooth CDFs. This further

Model	# $Tri/Quad$	δ	η	ζ	t_{cdf}	t_p	t_{PQ}
Airport	7214/2968	0.05°	0.023	0.002	2.9	12.5	10
Tower	6751/1447	0.05°	0.013	0.003	2.7	1.7	3.7
Costa	12202/2825	0.01°	0.012	0.005	4.8	34.2	11
Roof	10979/3536	0.05°	0.020	0.002	4.4	19	9
Shell	1653/635	0.01°	0.007	0.001	0.6	0.31	2
Bunny	28576/6920	0.08°	0.023	0.004	11.2	109.6	25

Table 1: Statistics and Timings. Timings are measured in seconds on a 2.66GHz Intel Quad core CPU with 8GB of RAM (our implementation utilizes multicores to parallelize the computation). From left to right: number of triangles and quads in the PQ mesh, face planarity δ , normalized diagonal distance η , perturbation distance ζ , the optimization time for a smooth CDF, the parametrization time and planarization time. The parametrization time varies with the number of triangles and singularities.

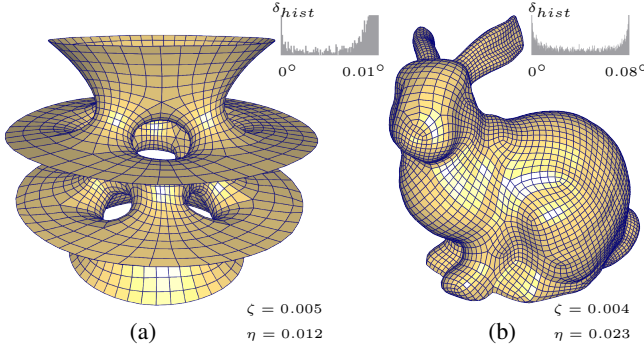


Figure 8: (a) The resulting PQ mesh of the Costa model. Note that our algorithm can generate a boundary conforming PQ mesh. (b) The PQ mesh of the Stanford Bunny.

shows the efficiency of our CDF optimization algorithm. The PQ mesh for the tower model contains vertices of valence 3, which corresponds to $1/4$ singularities in the CDF. This cannot be modeled in the representation vector based approach of [Zdravcevic et al. 2010].

Two more complicated results are shown in Figure 8. Figure 8a illustrates a PQ mesh for the Costa model with genus 6. The result shows that our algorithm can handle models of high genus. Moreover, by specifying strokes at the boundary and employing the similar feature-line-alignment technique in [Bommes et al. 2009], a boundary-conforming PQ mesh is obtained.

The Bunny model shown in Figure 6 is a challenging case for PQ mesh generation since the curvature tensor estimation is not faithful in some noisy regions. In this case, we relax the conjugacy condition by specifying $\epsilon_c = 0.1$ in the inequality constraint (Eqn. 8), and a pleasant PQ mesh is produced then (see Figure 8b).

Comparisons The superiority of the conjugate curve network over other curve networks in quad planarity has been proven in [Sauer 1970][Bobenko and Tsarev 2007, pp 10-11]. We briefly review their conclusion: taking four points $A = f(u_0, v_0)$, $B = f(u_0 + \epsilon, v_0)$, $C = f(u_0, v_0 + \epsilon)$, $D = f(u_0 + \epsilon, v_0 + \epsilon)$ from a smooth surface $f(u, v)$, the Euclidean distance $d(D, \pi_{ABC})$ from D to the plane spanned by A, B, C is $O(\epsilon^4)$ if and only if the u, v -curvilinear net is a conjugate net. If the conjugacy does not hold, $d(D, \pi_{ABC})$ is $O(\epsilon^3)$.

Fig. 9 illustrates this superiority using a simple cylinder model with radius 1. On the left of Fig. 9, we show the parameter lines of a parametrization: $f_\gamma(u, v) = (\cos v, \sin v, u \cos \gamma + v \sin \gamma)$, where γ is a parameter to control the angle between (u, v) param-

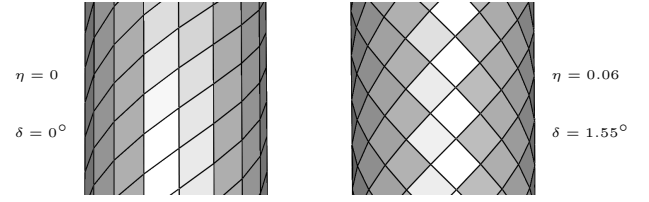


Figure 9: Two curve networks on a cylinder model. Left: a conjugate curve network. Right: a 4-RoS curve network. Note the quads from the conjugate curve network are exactly planar. Two curve networks are generated by setting the parameter γ to be $\pi/4$.

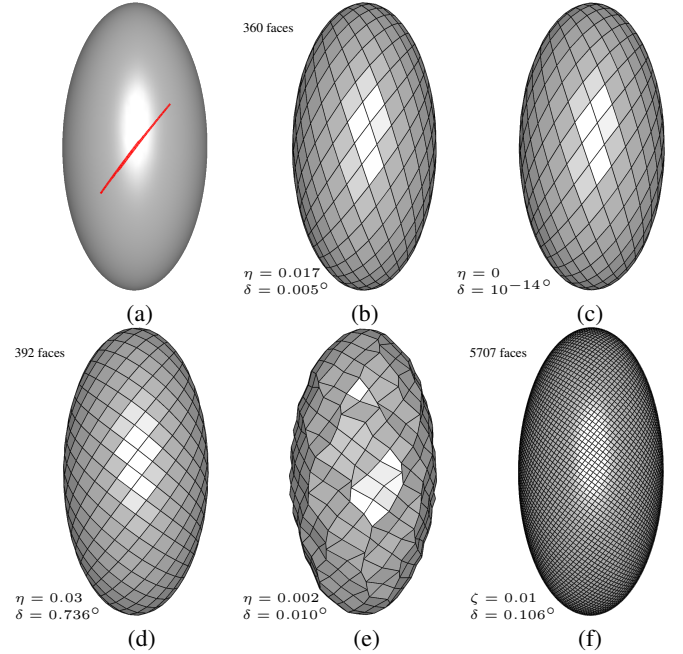


Figure 10: Comparison between CDF and 4-RoS field in the PQ mesh generation of a half ellipsoid. (a) a red stroke which specifies the directional constraint. (b&c) Quad meshes generated from our CDF design method before and after planarization. (d) A quad mesh generated by the mixed-integer method [Bommes et al. 2009] using a 4-RoS field. (e) The planarization result of (d). (f) A dense sampling of the quad mesh in (d).

eter lines. It characterizes a family of conjugate curve networks, since the u parameter lines are ruling directions on the cylinder. In this case, all the quads are exactly planar. On the right, the family of parameter lines are defined by the parametrization $g_\gamma(u, v) = (\cos(u \cos \gamma - v \sin \gamma), \sin(u \cos \gamma - v \sin \gamma), u \cos \gamma + v \sin \gamma)$. They are orthogonal curve networks, i.e., 4-RoS. However when $(\gamma \bmod \pi) \neq 0$, all the quads from the orthogonal network are not planar.

Since the initial quad mesh from a 4-RoS field might be far from planar, it can lead to a wrong local minimum in planarity optimization. Figure 10 illustrates an experimental verification of this situation. In Figure 10a, on a half ellipsoid model, a drawn stroke away from the principal curvature direction is specified to guide the generation of a PQ mesh. The meshes in Figure 10b and Figure 10d are generated by our CDF method and a 4-RoS field before planarization respectively. Although they have a similar number of faces, the planarity measure of our CDF result is one-order smaller since the orientation of quads are very different. We also compare the

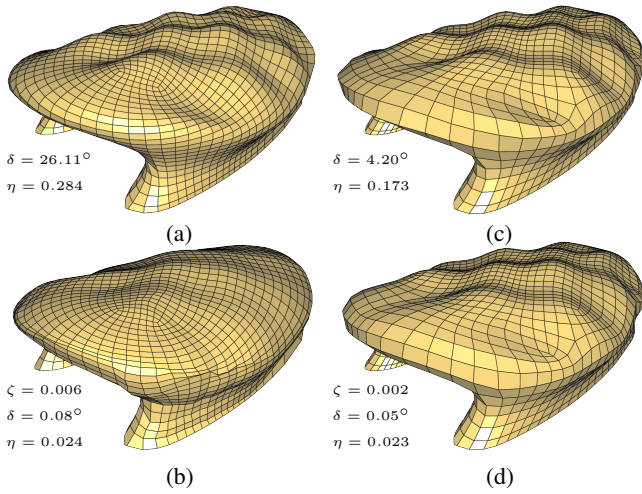


Figure 11: Comparison between PQ meshes from a 4-RoSy field and a CDF. (a&b) Quad meshes generated from a 4-RoSy field before and after planarization. (c&d) Quad meshes generated from our CDF before and after planarization.

qualities after planarity optimization. The mesh generated by our method can be perturbed slightly to be exactly planar. In contrast, a large distortion must be introduced to the quad mesh from 4-RoSy to improve its planarity since its parameter lines are far from conjugate. This validates the necessity of our CDF design method in PQ mesh generation. An alternative way to achieve similar planarity is to subdivide the mesh densely (see Figure 10e). However, increasing the number of faces is not desirable in architecture modeling, since it increases the fabrication cost significantly.

Figure 11 illustrates a comparison between a CDF and a 4-RoSy field in PQ mesh generation for the airport terminal model. Both fields are generated from the same set of user-specified strokes as shown in Figure 3. Since the conjugacy condition is not considered in the 4-RoSy field, the generated 4-RoSy field is far from conjugate although it contains fewer singularities (8 in total) than our CDF (24 in total). To achieve better planarity, the approximation quality is sacrificed (the bumps on the top are flattened in Figure 11-left). In contrast, the PQ mesh from our CDF design algorithm can approximate the original model faithfully.

A further validation of the advantage of CDF in PQ meshing is illustrated in Fig. 12. Comparing to the PQ mesh from CDF for the bunny model in Fig. 8b, the PQ mesh from the 4-RoSy field contains the distorted quads at the foot of the bunny model. The reason is that the 4-RoSy field cannot satisfy the conjugacy condition especially at high curvature regions. Therefore, the quad faces on these regions might be distorted with high probability after the planarization step (see Fig. 12b).

7 Discussion and Limitations

It is worth to mention that the usage of our method is not limited to architecture geometry. By removing the restriction of conjugacy and noticing that the conjugacy condition is not involved explicitly or implicitly in the smoothness measure in Section 3.1, our CDF is actually a general cross field where the angle does not need to be 90 degrees, so it may find applications in interactive modeling [Chen et al. 2008] and quad mesh generation [Bommes et al. 2010] to align the field with geometry. We believe that our formulation of the smoothness measure and index computation opens a door in designing a general cross field for more interesting computer graphics

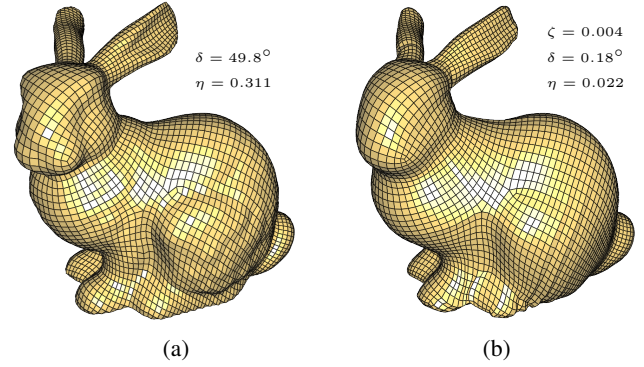


Figure 12: Quad meshes for the Bunny model generated from a 4-RoSy field before (a) and after (b) planarization.

applications.

There are a few limitations of our method that should be addressed. First, because of the nonlinearity of conjugacy, we cannot formulate the CDF optimization problem into a linear problem, which is different from the linear approaches in RoSy field design, such as the mixed integer approach in [Bommes et al. 2009] and the trivial connection in [Crane et al. 2010]. A proper initialization of the CDF must be provided for the success of the optimization. Figure 13 shows that a random initialization would introduce more singularities even after the CDF optimization. However, a random initialization is not intended in practice and a proper initialization has already been provided in Section 4.

The second limitation is the control of singularities. In our current algorithm, we cannot define the singularities in the design phase and a number of singularity editing operations is required for a complex model. It would be interesting to investigate how to incorporate the trivial connection approach into our CDF design for the user to exactly control the locus of the singularities.

8 Conclusion and Future Work

By introducing a novel signed-permutation-based representation of a smoothness measure for a CDF, we develop a general CDF design scheme in this paper. Since the vector association is treated as a permutation operation in this representation, arbitrary types of vector associations can be modeled to allow $\pm k/4$ singularities in a CDF. Furthermore the smoothness measure converts to a simple summation of cosine functions that simplifies the computation and results in an efficient direction field optimization algorithm.

In the future, we plan to integrate more functional properties into our CDF design to reduce the panel cost by limiting the types of quad faces [Fu et al. 2010] and to support the statics analysis for practical architectural construction [Schiftner and Balzer 2010]. Other research directions include the extension of the CDF to planar hexagonal meshes which possess many useful offset properties for fabrication [Pottmann et al. 2007b], and the study of discrete 3D conjugate nets [Bobenko and Suris 2008] inside a bounded volume for planar hexahedral mesh generation, where the planarity will improve the accuracy of linear hexahedral finite elements significantly.

Acknowledgements

Special thanks to Steve Lin for his careful proof-reading. The Bunny model is obtained courtesy of the Stanford 3D Scanning Repository. The airport, roof and tower model were created by Xi Zhang. Falai Chen is partially supported by National Basic

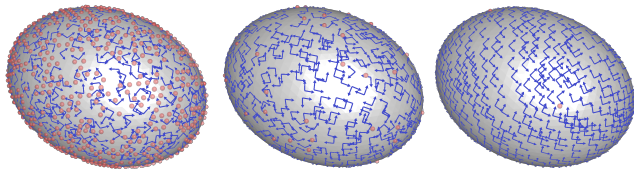


Figure 13: Left: a CDF is initialized randomly with singularities located at the red points. Middle: the optimized CDF by our method which contains 64 singular points. Right: with an initialization using the propagation approach in Section 4, the number of singularities can be reduced to 8.

Research Program of China (2011CB302400), and NSF of China (60873109, 11031007). Guoping Wang is partially supported by National Basic Research Program of China (2010CB328002) and NSF of China (60925007).

References

- ALLIEZ, P., COHEN-STEINER, D., DEVILLERS, O., LÉVY, B., AND DESBRUN, M. 2003. *Anisotropic polygonal remeshing*. *ACM Trans. Graph. (SIGGRAPH)* 22, 485–493.
- BOBENKO, A. I., AND SURIS, Y. B. 2008. *Discrete Differential Geometry*. American Mathematical Society.
- BOBENKO, A. I., AND TSAREV, S. P. 2007. *Curvature line parametrization from circle patterns*. arXiv:0706.3221.
- BOIER-MARTIN, I., RUSHMEIER, H., AND JIN, J. 2004. *Parameterization of triangle meshes over quadrilateral domains*. In *Symp. Geom. Proc.*, 193–203.
- BOMMES, D., ZIMMER, H., AND KOBBELT, L. 2009. *Mixed-integer quadrangulation*. *ACM Trans. Graph. (SIGGRAPH)* 28, 77:1–77:10.
- BOMMES, D., VOSSEMER, T., AND KOBBELT, L. 2010. *Quadrangular Parameterization for Reverse Engineering*. In *Mathematical Methods for Curves and Surfaces*. Springer Berlin / Heidelberg, 55–69.
- CHEN, G., ESCH, G., WONKA, P., MUELLER, P., AND ZHANG, E. 2008. *Interactive Procedural Street Modeling*. *ACM Trans. Graph. (SIGGRAPH)* 27, 3, 103:1–103:10.
- COHEN-STEINER, D., AND MORVAN, J.-M. 2003. *Restricted Delaunay Triangulations and Normal Cycle*. In *SoCG*, 312–321.
- CRANE, K., DESBRUN, M., AND SCHRÖDER, P. 2010. *Trivial Connections on Discrete Surfaces*. In *Comp. Graph. Forum (Symp. Geom. Proc.)*, vol. 29, 1525–1533.
- DO CARMO, M. 1976. *Differential Geometry of Curves and Surfaces*. Prentice-Hall.
- DONG, S., BREMER, P.-T., GARLAND, M., PASCUCCHI, V., AND HART, J. C. 2006. *Spectral surface quadrangulation*. *ACM Trans. Graph. (SIGGRAPH)* 25, 1057–1066.
- EIGENSATZ, M., KILIAN, M., SCHIFTNER, A., MITRA, N. J., POTTSMANN, H., AND PAULY, M. 2010. *Paneling architectural freeform surfaces*. *ACM Trans. Graph. (SIGGRAPH)* 29, 45:1–45:10.
- FU, C.-W., LAI, C.-F., HE, Y., AND COHEN-OR, D. 2010. *K-set tilable surfaces*. *ACM Trans. Graph. (SIGGRAPH)* 29, 44:1–44:6.
- GLYMPH, J., SHELDEN, D., CECCATO, C., MUSSEL, J., AND SCHOBER, H. 2004. *A parametric strategy for free-form glass structures using quadrilateral planar facets*. *Automation in Construction* 13, 187–202.
- HERTZMANN, A., AND ZORIN, D. 2000. *Illustrating smooth surfaces*. In *SIGGRAPH*, 517–526.
- HUANG, J., ZHANG, M., MA, J., LIU, X., KOBBELT, L., AND BAO, H. 2008. *Spectral quadrangulation with orientation and alignment control*. *ACM Trans. Graph.* 27, 147:1–147:9.
- KÄLBERER, F., MATTHIAS, N., AND POLTHIER, K. 2007. *Quad-Cover - Surface Parameterization using Branched Coverings*. *Comp. Graph. Forum (Symp. Geom. Proc.)* 26, 375–384.
- LIU, D. C., AND NOCEDAL, J. 1989. *On the Limited Memory Method for Large Scale Optimization*. *Mathematical Programming B* 45, 503–528.
- LIU, Y., POTTSMANN, H., WALLNER, J., YANG, Y.-L., AND WANG, W. 2006. *Geometric modeling with conical meshes and developable surfaces*. *ACM Trans. Graph. (SIGGRAPH)* 25, 681–689.
- NOCEDAL, J., AND WRIGHT, S. J. 1999. *Numerical Optimization*. Springer.
- PALACIOS, J., AND ZHANG, E. 2007. *Rotational symmetry field design on surfaces*. *ACM Trans. Graph. (SIGGRAPH)* 26, 55:1–55:10.
- POTTSMANN, H., AND WALLNER, J. 2008. *The focal geometry of circular and conical meshes*. *Adv. Comp. Math* 29, 249–268.
- POTTSMANN, H., ASPERL, A., HOFER, M., AND KILIAN, A. 2007. *Architectural Geometry*. Bentley Institute Press.
- POTTSMANN, H., LIU, Y., WALLNER, J., BOBENKO, A., AND WANG, W. 2007. *Geometry of Multi-layer Freeform Structures for Architecture*. *ACM Trans. Graph. (SIGGRAPH)* 26, 65:1–65:12.
- RAY, N., LI, W. C., LÉVY, B., SHEFFER, A., AND ALLIEZ, P. 2006. *Periodic global parameterization*. *ACM Trans. Graph.* 25, 1460–1485.
- RAY, N., VALLET, B., LI, W. C., AND LÉVY, B. 2008. *N-symmetry direction field design*. *ACM Trans. Graph.* 27, 10:1–10:13.
- RAY, N., VALLET, B., ALONSO, L., AND LEVY, B. 2009. *Geometry-aware direction field processing*. *ACM Trans. Graph.* 29, 1:1–1:11.
- SAUER, R. 1970. *Differenzengeometrie*. Springer.
- SCHIFTNER, A., AND BALZER, J. 2010. *Statics-Sensitive Layout of Planar Quadrilateral Meshes*. In *Advances in Architectural Geometry*.
- TONG, Y., ALLIEZ, P., COHEN-STEINER, D., AND DESBRUN, M. 2006. *Designing quadrangulations with discrete harmonic forms*. In *Symp. Geom. Proc.*, 201–210.
- ZADRAVEC, M., SCHIFTNER, A., AND WALLNER, J. 2010. *Designing Quad-dominant Meshes with Planar Faces*. *Comp. Graph. Forum (Symp. Geom. Proc.)* 29, 5, 1671–1679.