Remeshing II
Quad Remeshing

Generate a quad (or quad-dominant) mesh that approximates the input
Applications

• Fitting B-spline surfaces

• Simulation

• Subdivision

• Modeling

• Architecture
Requirements

• Lengths/angles distribution

• Orthogonality

• Alignment with curvature directions

• Regularity

• Planarity
Alignment with Curvature Directions

• More important than for triangular remeshing
  – Visible artifacts even for medium resolution

  – Aligned quads are closer to planar
Curvature Directions

Second fundamental form defines local orthogonal frame
Cross Fields

Min curvature  Max curvature  Cross Field
Cross Fields

Two orthogonal directions per triangle specify **orientation** of quads

Do not specify **sizing**
Umbilics

• Minimal curvature = maximal curvature
• No well defined directions locally
Cross Fields vs. Vector Fields

- Umbilics generate *singularities* in the cross field
- There is no consistent selection of 1 direction which gives a smooth vector field
Cross Fields vs. Vector Fields

- Umbilics generate **singularities** in the cross field
- There is no consistent selection of 1 direction which gives a smooth vector field
Singularities in the Wild

Index = -1/4

Index = 1/4

Index = 1/2

Index = -1/2
Methods

• **Trace curves** [Alliez ’03, Marinov ’04]
  – Generate curves tangent to cross field
  – Intersection of curves → vertices of quad mesh
  – T-Junctions, quad-dominant
Methods

• Base mesh [Dong ’06, Tong ’06, Huang ’08]
  – Construct coarse quad mesh
  – Refine it while preserving continuity
  – Hard to construct base mesh
    • harder than quad meshing
Methods

• Contouring [Ray ’06, Kalberer ’07, Zimmer ’09]
  – Find global parameterization \((u,v)\)
    • \(u\) and \(v\) can change roles!
  – Contour iso-\(u\), iso-\(v\) lines
Why is Quad Remeshing Hard?

• Local methods generate T-Junctions

• Base mesh is too restrictive
  – Some good quad meshes do not have a base mesh

• Can’t have both uniformity and alignment to curvature directions unless special surface

• Locations and valences of singular vertices have a global effect
Anisotropic Polygonal Remeshing

[Alliez et al. '03]
Anisotropic Polygonal Remeshing
[Alliez et al. ’03]
0. Compute Curvature Directions
3D Curvature Tensor

**Anisotropic**

- 2 principal directions

- \( k = 0 \)
- \( k_{min} < 0 \)
- \( k_{max} > 0 \)

- **Elliptic**
- **Parabolic**
- **Hyperbolic**

**Isotropic**

- \( k > 0 \)

- **Spherical**
- **Planar**

\( k \), \( k_{min}, \) and \( k_{max} \) are the components of the curvature tensor.
1. Flatten to 2D

- one 3D tensor per vertex
- discrete conformal parameterization
- piecewise linear interpolation of 2D tensors

2D tensor **field** using barycentric coordinates
2. Find Singularities

- Regular case

minor directions

major directions

Both directions
2. Find Singularities

- umbilic (spherical point)
- 2D tensor proportional to identity

- trisector
- wedge
Umbilics
3. Trace Curvature Lines

- Numerical integration of curvature directions

- Need to take care of:
  - Choosing seed points
  - Spacing of lines
  - Numerical stability
3. Trace Curvature Lines

- minor net
- major net
- overlay
3. Trace Curvature Lines

minor net

major net
4. Overlay

- Overlay curvature lines in anisotropic regions

- Add umbilic points in isotropic regions
5. Meshing

- **anisotropic areas (elliptic or hyperbolic)**
- **vertices (intersections)**
- **edges**
- **faces**

- **isotropic areas (spherical)**
- **vertices (points)**
- **edges (Delaunay)**
- **faces**
5. Meshing

Vertices

intersect lines of curvatures
5. Meshing

Edges

straighten lines of curvatures + Delaunay triangulation near umbilics
5. Meshing

Resolve T-Junctions
Close-up

minor net

remesh
Limitations

• Global parameterization
  • Smart cutting required
  • Low distortion parameterization
  • Numerical issues

• Lines of curvature
  • Robust curvature tensor computation
  • Tensor smoothing required
  • Optimal placement of streamlines difficult

• Contains non-quad elements
Global Parameterization with Cone Points

• Find global parameterization s.t. gradients align with input cross-field
  – Can be curvature directions, or any other

• Quad mesh “easy” to generate:
  – Take 2D grid covered by parameterization
QuadCover [Kalberer ’07]

The parameter function ... 

... has two gradient vector fields:

Integration of input fields yields parameterization.
Integrability

• Given $f$ find $g$ s.t. $f = \nabla g$

• Only possible if $f$ is "integrable"

• A vector field $U$ is *locally integrable* iff

  $\nabla \times U = 0$

  – No closed local loops!
Hodge-Helmholtz Decomposition

The space of vector fields on any surface decomposes into:

\[ X = \text{potential field } \nabla u + \text{curl-component } J \nabla u + \text{harmonic field } H \]

- Potential field $\nabla u$ is integrable.
- Curl-component $J \nabla u$ is not integrable.
- Harmonic field $H$ is locally integrable.
Assure Local Integrability

**Problem:** cross field \( K \) is usually not locally integrable

**Solution:** (assume frame \( K \) splits into two vector fields)

1. Compute Hodge-Helmholtz Decomposition \( K_1, K_2 \)
2. Remove curl-component (non-integrable part) of \( K_1, K_2 \)

\[
X_1 := K_1 - J\nabla v_1 = \nabla u_1 + J\nabla v_1 + H_1
\]
\[
X_2 := K_2 - J\nabla v_2 = \nabla u_2 + J\nabla v_2 + H_2
\]

**Result:** new cross field \( X = (X_1, X_2) \) is locally integrable
Global Integrability

**Problem:** mismatch of parameter lines around closed loops

**Solution:** mismatch of parameter lines around closed loops

1. Compute Homology generators (= basis of all closed loops)
2. Measure mismatch along Homology generator (next slide...)
Assure Global Integrability

Solution: (... cont’ed)

2. Measure mismatch along Homology generator $\gamma$ as curve integrals of both vector fields:

$$\int X_1 ds \in \mathbb{R}, \quad \int X_2 ds \in \mathbb{R}$$

3. Compute $L_2$-smallest harmonic vector fields $H_1, H_2$ s.t.

$$\int_{\gamma_p} (X_1 + H_1) ds \in \mathbb{Z}, \quad \int_{\gamma_p} (X_2 + H_2) ds \in \mathbb{Z}$$

Result: new frame $X = (X_1 + H_1, X_2 + H_2)$ is globally integrable
QuadCover Algorithm (unbranched)

Given a simplicial surface M:

1. Generate a guiding frame field $K$ 
   (e.g. principal curvatures frames)

2. Assure local integrability of $K$ via Hodge Decomp. 
   (remove curl-component from $K$)

3. Assure global continuity of $K$ along Homology gens. 
   (add harmonic field to $K$ s.t. all periods of $K$ are integers)

4. Global integration of $K$ on $M$ 
   gives parameterization
No Splitting of Parameter Lines

**Warning:** parameter lines do not split into red and blue lines !!!

**Consequence:** a frame field does not globally split into four vector fields.
Construct a Branched Covering Surface

**Step 1:** Make four layers (copies) of the surface.

**Step 2:** Lift frame field to a vector field on each layer.
Construct a Branched Covering Surface

**Step 3:** Connect layers consistently with the vectors.

**Result:** The frame field simplifies to a vector field on the covering surface.
Examples

Minimal surfaces with isolated branch points

Index of each singularity = $-1/2$

Trinoid

Schwarz-P Surface
Examples

Different Frame Fields

Non-orthogonal frame on hyperboloid

Non-orientable Klein bottle
Examples

Rocker arm test model
Mixed Integer Quadrangulation

[Zimmer ’09]