Re-Meshing Surfaces

- Given input mesh generate new mesh which is “better”
  - Element sizing
  - Element shape
- BUT is near (geometrically) to original surface
### Hausdorff Metric

Given two sets (surfaces) P and Q

\[
H_p(Q) = \max_p \min_q ||p,q|| \\
H(P,Q) = \max(H_q(P), H_p(Q))
\]

- Point to point
- On mesh approximate by
  - Measuring vertex to surface distance
  - Measuring vertex to vertex distance
- Expensive to compute
  - Public Domain Code: Metro

### How to (re)mesh surfaces?

- Can we apply Delaunay triangulation?
  - What is Delaunay criterion on surface?
    - Option 1: Use sphere instead of circle
      - Works for volumetric meshes (tets)
    - Option 2: Use pairwise test only
      - Theoretical Delaunay properties do not hold
  - Boundary recovery = Approximation quality
Approaches

- Mesh adaptation/Local Remeshing
  - Modify existing mesh using sequence of local operations
    - Evaluate approximation quality at every step

- Reduction to 2D/Global Remeshing
  - Segment surface into parameterizable pieces
  - Parameterize in 2D
  - Mesh in 2D (Delaunay)
  - Project back

Reduction to 2D/Global Remeshing

- Segment surface into parameterizable charts
  - distortion/chart size (count) trade-off
- Parameterize in 2D
  - Distortion affects 3D mesh quality
- Mesh charts in 2D (Delaunay)
  - Take parametric distortion into account (sizing)
  - Take care of shared boundaries
- Project back
Parameterization

- Projection to/from 2D should not distort mesh
- Can handle some stretch
  - Measure & take into account during 2D meshing
    - Use as component of local sizing
- MUST be conformal
  - If care about quality
  - Want: Equilateral (Delaunay) in 2D = equilateral (Delaunay) in 3D

Impact of distortion
Segmentation

- Chart Properties
  - parameterizable: open + genus 0
  - Low distortion
    - Ideal: Developable charts

- Approaches
  - Single chart
    - Generate (short) cuts to reduce genus
    - Cut through high curvature/distortion vertices
  - Multiple charts
    - More convex boundaries - easier to handle

Lloyd Segmentation Framework

- Lloyd iterations:
  - Select random triangles to act as seeds
  - Grow charts around seeds using a greedy approach
  - Find new seed for each chart
    - Typically chart center
  - Repeat from step 2 until convergence
Proxies

- Charts represented by proxies – used for reseeding and growth
- Example I: Planar charts
  - Proxy: Normal to plane $N_c$
  - Compute: Average normal of chart triangles
  - Growth metric: Normal difference $F(C,t) = N_c \cdot n_t$

Example II: D-Charts (Developmental Charts)

- Constant angle between surface normal and axis → Developable chart
- Proxy: $\langle \text{axis, angle} \rangle = \langle N_c, \theta_c \rangle$
- Compute:
  $\min_{N_c \in A_c} \frac{1}{A_c} \sum_{t \in C} A_f(C,t)$ s.t. $\|N_c\| = 1$
- Growth metric:
  $F(C,t) = (N_c \cdot n_t - \cos \theta_c)^2$
- Combine with compactness
Examples

Example Results
Meshing - sizing

- Measure parametric stretch (3D to 2D)
- Measure stretch per edge $||e_{3D}||/||e_{2D}||$
- Vertex stretch = average of edges
- Multiply sizing function (at vertices) by stretch

Example (use WCVD)
Boundary

- Need mesh consistency along boundaries
- Enforce shared boundary vertex positions

Boundaries

- Consistent but visible...
Features

- Preserving features - locate surface creases and prevent removing them
  - Special handling by segmentation and/or 2D meshing

Global Methods - Properties

- Three major components:
  - Segment
  - Parameterize
  - Mesh in 2D

- Strongly depends on parameterization quality
  - In turn depends on segmentation
  - More “formal”

- Typically more complex to implement from scratch