Geometry Processing

4 Parameterization

Ludwig-Maximilians-Universität München
Session 4: Parameterization

● Motivation

● Methods
  ○ Tutte's Embedding Theorem (Barycentric Mapping)
  ○ Least Squares Conformal Maps (L SCM)
  ○ Angle-based Flattening (ABF)

● Summary

● Discussion: Parameterization in Blender
Parameterization: Definition

A function $f$ that maps input surface in **one-to-one** correspondence with a different (e.g. 2D) domain

Example: In UV Mapping:
- Each vertex is associated with an UV coordinate $(x_i, y_i, z_i) \rightarrow (u_i, v_i)$
- Caution: vertices at seam

Equivalent terminologies: Flattening, unfolding
Parameterization: Applications

Different types:

- Surface to plane mapping
  - Producing UV/normal/displacement/... maps
  - Compression [Gu et al 2002] (UE5’s Virtual Geometry)
  - ...

- Plane to surface mapping
  - Remeshing (later)
  - ...

- Surface to surface mapping
  - Deformation (later)
Session 4: Parameterization

- Motivation
- Methods
  - Tutte's Embedding Theorem (Barycentric Mapping)
  - Least Squares Conformal Maps (LSCM)
  - Angle-based Flattening (ABF)
- Summary
- Discussion: Parameterization in Blender
Barycentric Mapping (Tutte's Embedding Theorem [Tutte 1960])

From graph theory: Given a triangulated surface homeomorphic to a disk, if the \((u, v)\) coordinates at the boundary vertices lie on a convex polygon in order, and if the coordinates of the internal vertices are a convex combination of their neighbors, then the \((u, v)\) coordinates form a valid parameterization.

A convex polygon: circle, square, ...

Interior vertices: \(\{1, \ldots, n_{\text{int}}\}\)  Boundary vertices: \(\{n_{\text{int}} + 1, \ldots, n\}\)

A convex combination (barycentric coordinates): 

\[
-w_{ii} \begin{pmatrix} u_i \\ v_i \end{pmatrix} = \sum_{j \neq i} w_{ij} \begin{pmatrix} u_i \\ v_j \end{pmatrix}
\]
Barycentric Mapping: Matrix Form

For interior vertices:

\[-w_{ii} \begin{pmatrix} u_i \\
v_i \end{pmatrix} = \sum_{j \neq i} w_{ij} \begin{pmatrix} u_i \\
v_j \end{pmatrix} \Rightarrow w_{ii} = -\sum_{i \neq j} w_{ij} \]

Recall Laplace matrix:

\[
\mathbf{L} = \mathbf{D} \mathbf{W} \\
\mathbf{W} = (W_{ij}) \\
W_{ij} = \begin{cases} -\sum_{i_k} w_{i_k}, & \text{if } i = j \\ w_{ij}, & \text{if } j \text{ is a neighbor of } i \\ 0, & \text{otherwise} \end{cases}
\]

All we need is to solve two linear equations: \( \mathbf{L} \mathbf{u}' = \mathbf{u} \quad \mathbf{L} \mathbf{v}' = \mathbf{v} \)

where the elements of \( \mathbf{u} \) (respectively \( \mathbf{v} \)) is either zero (interior vertices) or precomputed (boundary vertices)

The solution \((\mathbf{u}', \mathbf{v}')\) is the barycentric mapped UV coordinates.
Barycentric Mapping: Uniform Laplacian as Example

\[ L = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 1 & -5 & 1 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 1 & -5 & 1 \\
0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & -5 
\end{pmatrix} \quad u' = \begin{pmatrix}
u_1 \\
u_2 \\
u_3 \\
u_4 \\
u_5 \\
u_6 \\
u_7 \\
u_8 \\
u_9 
\end{pmatrix} \quad u = \begin{pmatrix}
u_{v1} \\
u_{v2} \\
u_{v3} \\
u_{v4} \\
u_{v5} \\
u_{v6} \\
0 \\
0 \\
0 
\end{pmatrix} \quad Lu' = u \quad Lv' = v
Barycentric Mapping: Intuition & Issues

Intuitively:

Different choice of Laplace matrix: Uniform, Cotan, ...

**Caution:** Tutte's embedding theorem requires the Laplacian to satisfy: LOC+LIN+POS *(why?)*

Cotan Laplacian can violate POS, there is a better version "mean value weights" [Floater 03] produce provably positive ones:

\[ w_{ij} = \frac{1}{\| f_i - f_j \|} \left( \tan \left( \frac{\gamma_{ij}}{2} \right) + \tan \left( \frac{\delta_{ij}}{2} \right) \right) \]
Barycentric Mapping: Issues (cont.)

Tutte's Embedding requires a fixed convex boundary ⇒ High distortion and mesh must have at least a boundary

How to minimize the distortion?
How to cut a "watertight" mesh?
How to achieve free boundary?
Texture Atlas Generation [Lévy et al 2002]

The generation of a texture atlas can be decomposed into the following steps:

1. **Segmentation**: The model is partitioned into a set of charts
2. **Parameterization**: Each chart is ‘unfolded’, i.e. put in correspondence with a subset of $\mathbb{R}^2$
3. **Packing**: The charts are gathered in texture space.

This workflow still largely exists in today's modeling practice (either manual mark seam or "smart" unwrap)
Conformal Mapping

Conformal mappings = rotation + scale

Scale distortion is smoothly distributed (harmonic)

Equivalent terminologies: Angle-preserving, similar, scale
Mapping in General

\[ f(u + \Delta u, f + \Delta v) = f(u, v) + f_u(u, v)\Delta u + f_v(u, v)\Delta v \Rightarrow f(u + \Delta u, v + \Delta v) = f(u, v) + J\begin{pmatrix} \Delta u \\ \Delta v \end{pmatrix} \]

To achieve angle preservation, the Jacobian must be a similarity transformation:

\[
J = \begin{pmatrix}
\frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\
\frac{\partial v}{\partial x} & \frac{\partial v}{\partial y}
\end{pmatrix} = s \begin{pmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix}
\frac{\partial u}{\partial x} \\
\frac{\partial u}{\partial y}
\end{pmatrix} = \begin{pmatrix}
\frac{\partial v}{\partial y} \\
-\frac{\partial v}{\partial x}
\end{pmatrix}
\]

(Surprisingly leads to Cauchy-Riemann Equations)
Least Squares Conformal Maps (LSCM, ASAP) [Lévy et al 2002]

If the parameterization subject to
\[
\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y}, \quad \frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}
\]
then it is a conformal, but in general it is impossible (why?)

Instead, we minimize the least square "energy" as our objective:

\[
E_{\text{LSCM}} = \sum_t A_t \left( \left( \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} \right)^2 + \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 \right)
\]

(triangle area)

The energy measures non-conformality

It is invariant with respect to arbitrary translations and rotations
Least Squares Conformal Maps (LSCM, ASAP) [Lévy et al. 2002]

Issues:
Energy does not have a unique minimizer, one can fix at least two vertices.
The choice of the vertices affects the results significantly
No guarantee on bijective
No guarantee on flip-free
...
Angle-based Flattening (ABF)

More straightforward: Given angles for original mesh, find the closest angle that describe a flat mesh

$$\min_{\theta} \sum_i (\hat{\theta}_i - \theta_i)^2$$

Subject to

1. angle sum: \( \theta_i + \theta_j + \theta_k = \pi \)

2. interior vertices sum: \( \sum_{ijk} \theta_i = 2\pi \)

3. compatible lengths around vertices (law of sines): \( \prod_{ijk} \frac{\sin \theta_j}{\sin \theta_k} = 1 \)

Nonlinear optimization, many approximations: LinearABF [Zayer et al 2007], ABF++ [Sheffer et al 2005], etc.

You have the ability to known how they work exactly and be able to implement them eventually.
Session 4: Parameterization

● Motivation

● Methods
  ○ Tutte's Embedding Theorem (Barycentric Mapping)
  ○ Least Squares Conformal Maps (LSCM)
  ○ Angle-based Flattening (ABF)

● Summary

● Discussion: Parameterization in Blender
Summary

- Texture atlas generation workflow containing three parts: Segmentation, parameterization, and packing
- Tutte's embedding is a classic baseline for parameterization
- Objectives: Minimize distortion (any kind of properties) with respect to a certain measure
  - Validity (bijective, no self intersection): flipped triangle?
  - Boundary: fixed or free?
  - Domain: 2D or 3D?
  - ...
- A very large number of techniques exists that solves different problems (no best solution!)
Further Readings


Session 4: Parameterization

● Motivation

● Methods
  ○ Tutte's Embedding Theorem (Barycentric Mapping)
  ○ Least Squares Conformal Maps (LSCM)
  ○ Angle-based Flattening (ABF)

● Summary

● Discussion: Parameterization in Blender
Mesh Parameterization in Blender

User Manual

- https://docs.blender.org/manual/en/latest/modeling/meshes/editing/uv.html#unwrap
- https://docs.blender.org/manual/en/latest/modeling/meshes/editing/uv.html#smart-uv-project

API Manual

- https://docs.blender.org/api/current/bpy.ops.uv.html?highlight=uv#bpy.ops.uv.unwrap

Implementation

- source/blender/editors/uvedit/uvedit_parameterizer.c (c2a01a6c118e)
Blender: Geometry Nodes

Beta release: https://builder.blender.org/download/


Workboard: https://developer.blender.org/project/board/121/
Dimensions: A walk through mathematics (2011)

https://www.youtube.com/watch?v=yJZP_-40KVw&list=PL97CCC2CC4E89C7E5

https://www.youtube.com/watch?v=MWHMzgZ4Vlk