Surface Parameterization
Maks Ovsjanikov
Raif Rustamov
Justin Solomon
Mirela Ben-Chen
Julien Tierny
SIGGRAPH 2008 Course
Others…
Today

- Painting on Surfaces

Painting directly on the 3d object

*Mari software*  
*Substance 3D Painter*
What we would like to do

Input geometry → Packing

Texture map
The Basic Problem
Solution
Parameterization is ...
Why Parameterize?
Why Parameterize?

Texture Mapping

http://www.blender.org/development/release-logs/blender-246/uv-editing/

Texture Mapping
Parameterization Problem

Given a surface (mesh) $S$ in $\mathbb{R}^3$ and a domain $\Omega$ (e.g. plane):
Find a bijective map $U: \Omega \leftrightarrow S$. 
Parameterization for Texture Mapping
Parameterization for Texture Mapping

Rendering workflow:

Models and texture maps from Poser, a product for creating and rendering human characters.
Parameterization – Typical Domains

disk = genus zero + boundary

sphere = closed genus zero
Parameterization – Boundary Problem

Source: Mirela Ben-Chen
Parameterization – Many Possibilities

Source: Mirela Ben-Chen
Recall Mesh simplification:
- Approximate the geometry using few triangles

Idea:
- Decouple geometry from appearance
Parameterization – Applications

Recall Mesh simplification:
  • Approximate the geometry using few triangles

Idea:
  • Decouple geometry from appearance

Observation: appearance (light reflection) depends on the geometry + normal directions.
Parameterization – Applications

Normal Mapping

Idea:
• Decouple geometry from appearance
• Encode a normal field inside each triangle
Normal Mapping with parameterization:

- Store normal field as an RGB texture.
Parameterization – Applications

- Remeshing

source: Mirela Ben-Chen
Parameterization – Applications

source: Mirela Ben-Chen
Parameterization – Applications

- Compression

Stanford Bunny

Gu, Gortler, Hoppe. Geometry Images. SIGGRAPH 2002
Parameterization – Applications

General Idea: Things become easier in a canonical domain (e.g. on a plane).

Other Applications:
- Surface Fitting
- Editing
- Mesh Completion
- Mesh Interpolation
- Morphing and Transfer
- Shape Matching
- Visualization
...
Parameterization onto the plane

General problem:

- Given a mesh \((T, P)\) in 3D find a bijective mapping

\[
g : P \rightarrow \mathbb{R}^2
\]

\[
g(p_i) = u_i = (u_i, v_i)
\]
Parameterization onto the plane

General problem:
- Given a mesh \((T, P)\) in 3D find a bijective mapping

\[
g : P \rightarrow \mathbb{R}^2
\]

\[
g(p_i) = u_i = (u_i, v_i)
\]
Parameterization onto the plane

**Simplified problem:**
- Given a mesh \((T, P)\) in 3D find a bijective mapping

\[
g : P \rightarrow \mathbb{R}^2
\]

\[g(p_i) = u_i = (u_i, v_i)\]

under some boundary constraints:

\[g(b_j) = u_j \text{ for some } \{b_j\}\]
Parameterization onto the plane

Recall a related problem.

**Mapping the Earth**: find a parameterization of a 3d object onto a plane.
Mapping the earth

Stereographic projection

Maps circles to circles

Hipparchus (190–120 B.C.)
Mapping the earth

Mercator

Maps loxodromes to lines

Gerardus Mercator (1569)
Mapping the earth

Mercator (preserves angles, but distorts areas)

Maps loxodromes to lines

Gerardus Mercator (1569)
SIZE OF RUSSIA COMPARED TO AFRICA

Africa is almost twice the size of Russia.

Africa: 11.72 million sq. mi.
Africa population: 1.11 billion

Russia: 6.593 million sq. mi.
Russia population: 143 million
Mapping the earth

http://thetruesize.com
The size of Westeros compared to the USA

http://thewertzone.blogspot.com/2012/06/size-of-westeros-compared-to-usa.html
Mapping the earth

Lambert (preserves areas, but distorts angles)

Johann Heinrich Lambert (1772)
Mapping the earth

Lambert (preserves areas, but distorts angles)

Johann Heinrich Lambert (1772)
Different kinds of Parameterization

Various notions of distortion:

1. Equiareal: preserving areas (up to scale)
2. Conformal: preserving angles of intersections
3. Isometric: preserving geodesic distances (up to scale)

**Theorem**: Isometric = Conformal + Equiareal
Different kinds of Parameterization

**Intrinsic properties:**
Those that depend on angles and distances on the surface. E.g.

- **Intrinsic:** geodesic distances
- **Extrinsic:** coordinates of points in space

**Remark:**
Intrinsic properties are preserved by isometries.

**Bad news:**
Gauss’s Theorema Egregium: curvature is an intrinsic property.
There is no isometric mapping between a sphere and a plane.
Different kinds of Parameterization

- **Orthographic**
  - Preserves angles = **conformal**

- **Stereographic**

- **Mercator**
  - Preserves area = **equiareal**

- **Lambert**
Different kinds of Parameterization
Different kinds of Parameterization

Since we are dealing with a triangle mesh, we first need to ensure a bijective map
Given a mesh \((T, P)\) in 3D find a bijective mapping \(g(p_i) = u_i\) given constraints: \(g(b_j) = u_j\) for some \(\{b_j\}\).

Model: imagine a **spring** at each edge of the mesh. If the boundary is fixed, let the interior points find an **equilibrium**.
Recall: potential energy of a spring stretched by distance $x$:

$$E(x) = \frac{1}{2} kx^2$$

$k$: spring constant.
Spring Model for Parameterization

Given an embedding (parameterization) of a mesh, the potential energy of the whole system:

\[ E = \sum_e \frac{1}{2} D_e \| u_{e1} - u_{e2} \|^2 \]

\[ = \frac{1}{2} \sum_{i=1}^{n} \sum_{j \in N_i} \frac{1}{2} D_{ij} \| u_i - u_j \|^2 \]

Where \( D_e = D_{ij} \) is the spring constant of edge \( e \) between \( i \) and \( j \)

Goal: find the coordinates \( \{ u_i \} \) that would minimize \( E \).

Note: the boundary vertices prevent the degenerate solution.
Parameterization with Barycentric Coordinates

Finding the optimum of:

\[ E = \frac{1}{2} \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \frac{1}{2} D_{ij} \| u_i - u_j \|^2 \]

\[ \frac{\partial E}{\partial u_i} = 0 \Rightarrow \sum_{j \in \mathcal{N}_i} D_{ij} (u_i - u_j) = 0 \]

\[ \Rightarrow u_i = \sum_{j \in \mathcal{N}_i} \lambda_{ij} u_j, \text{ where } \lambda_{ij} = \frac{D_{ij}}{\sum_{j \in \mathcal{N}_i} D_{ij}} \]

i.e. each point \( u_i \) must be an **convex combination** of its neighbors.
Hence: barycentric coordinates.
Parameterization with Barycentric Coordinates

To find the solution in practice:

1. Fix the boundary points \( b_i, i \in B \)
2. Form linear equations

\[
\begin{align*}
    & u_i = b_i, \quad \text{if } i \in B \\
    & u_i - \sum_{j \in N_i} \lambda_{ij} u_j = 0, \quad \text{if } i \notin B
\end{align*}
\]

1. Assemble into two linear systems (one for each coordinate):

\[
LU = \bar{U}, \quad LV = \bar{V} \quad L_{ij} = \begin{cases} 
    1 & \text{if } i = j \\
    -\lambda_{ij} & \text{if } j \in N_i, \ i \notin B \\
    0 & \text{otherwise}
\end{cases}
\]

1. Solution of the linear system gives the coordinates:

   Note: system is very sparse, can solve efficiently. \( u_i = (u_i, v_i) \)
Parameterization with Barycentric Coordinates

Does this work?

- **Theorem (Maxwell-Tutte)**

  If $G = \langle V,E \rangle$ is a 3-connected planar graph (triangular mesh) then any **barycentric** drawing is a valid embedding.
Laplacian Matrix

Our system of equations (forgetting about boundary):

\[ u_i = \sum_{j \in \mathcal{N}_i} \lambda_{ij} u_j, \quad \text{where} \quad \lambda_{ij} = \frac{D_{ij}}{\sum_{j \in \mathcal{N}_i} D_{ij}} \]

\[ LU = 0 \quad L_{ij} = \begin{cases} 
1 & \text{if } i = j \\
-\lambda_{ij} & \text{if } j \in \mathcal{N}_i \\
0 & \text{otherwise}
\end{cases} \quad L \text{ is not symmetric} \]

Alternatively, if we write it as:

\[ u_i \sum_{j \in \mathcal{N}_i} D_{ij} = \sum_{j \in \mathcal{N}_i} D_{ij} u_j \]

We get:

\[ LU = 0 \quad L_{ij} = \begin{cases} 
\sum_{k \in \mathcal{N}_i} D_{ij} & \text{if } i = j \\
-D_{ij} & \text{if } j \in \mathcal{N}_i \\
0 & \text{otherwise}
\end{cases} \quad L \text{ is symmetric} \]
Parameterization with Barycentric Coordinates

Example:

Uniform weights:

$$D_{ij} = 1$$

Laplacian Matrix

$$w = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\
1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{pmatrix}$$
Alternative: Simple realization

Goal: Assign \((u, v)\) coordinate to each mesh vertex.

1. Fix \((u, v)\) coordinates of boundary.
2. Want interior vertices to be at the center of mass of neighbors:

\[
\begin{align*}
\hat{u}_i &= \frac{1}{|N(i)|} \sum_{j \in N(i)} u_j \\
\hat{v}_i &= \frac{1}{|N(i)|} \sum_{j \in N(i)} v_j
\end{align*}
\]
Iterative Algorithm

1. Fix \((u, v)\) coordinates of boundary.
2. Initialize \((u, v)\) of interior points (e.g. using naïve).
3. While not converged: for each interior vertex, set:

\[
\begin{align*}
    u_i & \leftarrow \frac{1}{|N(i)|} \sum_{j \in N(i)} u_j \\
    v_i & \leftarrow \frac{1}{|N(i)|} \sum_{j \in N(i)} v_j
\end{align*}
\]

\[
\begin{align*}
    u_1 & \leftarrow \frac{u_2 + u_3 + u_4 + u_5 + u_6 + u_7}{6} \\
    v_1 & \leftarrow \frac{v_2 + v_3 + v_4 + v_5 + v_6 + v_7}{6}
\end{align*}
\]
What do you think?

Some random planar mesh

After many iterations
Expectation

After many iterations

It is already planar: best parameterization = itself
Reality... Why? How to avoid?

After many iterations

Converges to a somewhat uniform grid!

Triangle shapes and sizes are not preserved!
Parameterization with Barycentric Coordinates

Linear Reproduction:
- If the mesh is already **planar** we want to recover the original coordinates.

Problem:
- Uniform weights do not achieve linear reproduction
- Same for weights proportional to distances.
Linear Reproduction:
- If the mesh is already **planar** we want to recover the original coordinates.

Problem:
- Uniform weights do not achieve linear reproduction
- Same for weights proportional to distances.

Solution:
- If the weights are **barycentric** with respect to original points:

\[ p_i = \sum_{j \in \mathcal{N}_i} \lambda_{ij} p_j, \quad \sum_{j \in \mathcal{N}_i} \lambda_{ij} = 1 \]

The resulting system will recover the planar coordinates.
Parameterization with Barycentric Coordinates

Solution:
- Barycentric coordinates with respect to original points:

\[ p_i = \sum_{j \in \mathcal{N}_i} \lambda_{ij} p_j, \quad \sum_{j \in \mathcal{N}_i} \lambda_{ij} = 1 \]

- If a point \( p_i \) has 3 neighbors, then the barycentric coordinates are **unique**.
- For more than 3 neighbors, many possible choices exist.
Some good news.

Riemann Mapping Theorem:

Any surface topologically equivalent to a disk, **can be** conformally mapped to a unit disk.

Cauchy-Riemann equations:

If a map \((x,y) \rightarrow (u,v)\) is conformal, then \(u(x,y)\) and \(v(x,y)\) satisfy:

\[
\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y}, \quad \frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}.
\]
Conformal Mappings

Some good news.

**Riemann Mapping Theorem:**

Any surface topologically equivalent to a disk, can be conformally mapped to a unit disk.

**Cauchy-Riemann equations:**

If a map \((x,y) \rightarrow (u,v)\) is conformal then both \(u\) and \(v\) are harmonic:

\[
\left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) u = 0
\]

\[
\left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) v = 0
\]
Conformal Mappings

Some good news.

**Riemann Mapping Theorem:**

Any surface topologically equivalent to a disk, **can be** conformally mapped to a unit disk.

**Cauchy-Riemann equations:**

If a map \((x,y) \rightarrow (u,v)\) is conformal then both \(u\) and \(v\) are harmonic:

\[ \Delta u = 0 \]
\[ \Delta v = 0 \]
Conformal Mappings

Some good news.

**Riemann Mapping Theorem:**

Any surface topologically equivalent to a disk, **can be** conformally mapped to a unit disk.

If a map \( S \rightarrow (u,v) \) is conformal then both \( u \) and \( v \) are harmonic:

\[
\Delta_S u = 0
\]

\[
\Delta_S v = 0
\]

\( \Delta_S \) : Laplace-Beltrami operator.
Harmonic Mappings

Recap:

Isometric $\Rightarrow$ Conformal $\Rightarrow$ Harmonic

Harmonic mappings easiest to compute, but may not preserve angles. May not be bijective.

Harmonic maps minimize Dirichlet energy:

$$E_D(f) = \frac{1}{2} \sum_S \| \nabla_S f \|^2$$

Given the boundary conditions.
Harmonic Mappings

Theorem (Rado-Kneser-Choquet):

If \( f : S \to R^2 \) is harmonic and maps the boundary \( \partial S \) onto the boundary \( \partial S^* \) of some convex region \( S^* \subset R^2 \), then \( f \) is bijective.
Recall the General Method:

To find the solution in practice:

1. Fix the boundary points \( b_i, i \in B \)

1. Assemble \textit{two} linear systems (one for each coordinate):

\[
LU = \bar{U}, \quad LV = \bar{V}
\]

where

\[
L_{ij} = \begin{cases} 
1 & \text{if } i = j, \ i \in B \\
\sum_{j \in N_i} D_{ij} & \text{if } i = j, \ i \notin B \\
-D_{ij} & \text{if } j \in N_i, \ i \notin B \\
0 & \text{otherwise}
\end{cases}
\]

1. Solution of the linear system gives the coordinates: \( u_i = (u_i, v_i) \)
Barycentric Coordinates: Harmonic

\[
D_{ij} = \frac{\cot(\alpha_{ij}) + \cot(\beta_{ij})}{2}
\]

- Weights can be negative – not always valid
- Weights depend only on angles - close to conformal
- 2D reproducible
Barycentric Coordinates: Mean-value

\[ D_{ij} = \frac{\tan(\gamma_{ij} / 2) + \tan(\delta_{ij} / 2)}{2 \| V_i - V_j \|} \]

- Result visually similar to harmonic
- No negative weights – **always** valid
- 2D reproducible
Results

Naive

Harmonic
Barycentric Coordinates

uniform

harmonic

mean-value
Conformal Mappings

Most commonly used in practice.
Conformal Mappings

Fixing the boundary:

- Simple convex shape (triangle, square, circle)
- Distribute points on boundary
  - Use chord length parameterization
- Fixed boundary can create high distortion
Conformal Mappings

Fixing the boundary:

• Simple convex shape (triangle, square, circle)
• Distribute points on boundary
  – Use chord length parameterization
• Fixed boundary can create high distortion

“Free” boundary is better: harder to optimize for.
Fixed vs Free boundary

images by Mirela Ben-Chen
Fixed vs Free boundary

images by Mirela Ben-Chen
Fixed vs Free boundary

images by Mirela Ben-Chen
Fixed vs Free boundary

images by Mirela Ben-Chen
Free boundary methods

General approach:

Let the coordinates of the vertices be unknowns, construct an energy that measures distortion.

\[ E(f) = \sum_{t \in \text{triangles}} \text{distortion}(t|f) \]

\[
(u_{\text{opt}}, v_{\text{opt}}) = \arg \min_{f=(u,v)} E(f) \quad \text{given boundary conditions}
\]
For any triangle:

\[(u_3, v_3) - (u_1, v_1) = \frac{\sin \alpha_2}{\sin \alpha_3} R^{\alpha_1} [(u_2, v_2) - (u_1, v_1)]\]

If the mapping is conformal, the angles shouldn’t change. Keep the angles, let the coordinates be unknown. Leads to a least squares problem.
Free boundary methods

More generally:

$$E(f) = \sum_{t \in \text{triangles}} \text{distortion}(t|f)$$

$$\text{distortion}(t|f) = H(J_f(t))$$

$$J_f(t) : \text{Jacobian of the transformation}$$
Free boundary methods

More generally:

\[ \text{distortion}(t|f) = H(J_f(t)) \]

\[ J_f(t) : \text{Jacobian of the transformation} \]

\[ J_f(t) = U \Sigma V^T = U \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{pmatrix} V^T \]

1. Isometric mapping: \( \sigma_1 = \sigma_2 = 1 \)
2. Conformal mapping: \( \frac{\sigma_1}{\sigma_2} = 1 \)
3. Equiareal mapping: \( \sigma_1 \sigma_2 = 1 \)
Free boundary methods

More generally:

\[ \text{distortion}(t|f) = H(J_f(t)) \]

\( J_f(t) \): Jacobian of the transformation

\[ J_f(t) = U\Sigma V^T = U \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{pmatrix} V^T \]

\[ H(J_f(t)) = H(\sigma_1, \sigma_2), \text{ e.g.:} \]

\[ H_{\text{MIPS}}(\sigma_1, \sigma_2) = \frac{\sigma_1}{\sigma_2} + \frac{\sigma_2}{\sigma_1} \]

Non-linear, difficult to optimize for.

MIPS: An efficient global parameterization method, Hormann and Greiner, Curve and Surface design, ‘99
Free boundary methods

More generally:

\[ \text{distortion}(t|f) = H(J_f(t)) \]

\[ J_f(t) : \text{ Jacobian of the transformation} \]

\[ J_f(t) = U \Sigma V^T = U \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{pmatrix} V^T \]

Can show that:

\[ \sigma_1^2 + \sigma_2^2 \quad \text{and} \quad \sigma_1 \sigma_2 \quad \text{are quadratic in the target vertex coordinates.} \]

Thus, e.g. \( H(\sigma_1, \sigma_2) = (\sigma_1 - \sigma_2)^2 \) leads to a linear system of equations.
Free boundary methods

More generally:

\[
\text{distortion}(t|f) = H(J_f(t))
\]

\(J_f(t)\) : Jacobian of the transformation

\[
J_f(t) = U\Sigma V^T = U \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{pmatrix} V^T
\]

Can show that:

\[
\sigma_1^2 + \sigma_2^2 \quad \text{and} \quad \sigma_1\sigma_2\]

are \textit{quadratic} in the target vertex coordinates.

Thus, e.g. \(H(\sigma_1, \sigma_2) = (\sigma_1 - \sigma_2)^2\) leads to a linear system of equations.
Some results

Linear Methods:

mean value

conformal
Some results

Non-linear Methods:

ABF++  circle patterns  MIPS  stretch
Conclusions

Surface parameterization:

- No perfect mapping method
- A very large number of techniques exists
- Conformal model:
  - Nice theoretical properties
  - Leads to a simple (linear) system of equations
  - Closely related to the Poisson equation and Laplacian operator
- More general methods
  - Can get smaller distortion using non-linear optimization
  - Very difficult to guarantee bijectivity in general
Breathing Type: Normal

Comparing Real vs. Animated Breathing