# Parameterization with Manifolds

Cindy Grimm

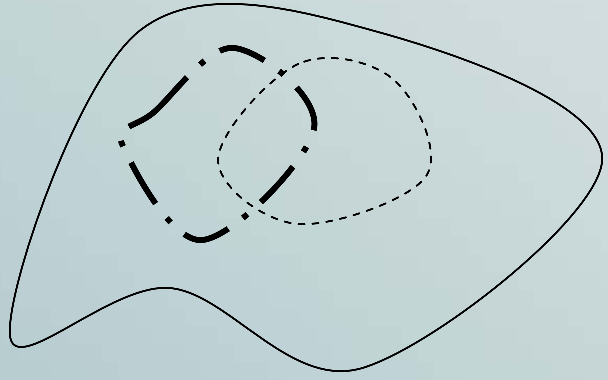
# Manifold

Media and Machines

- What they are
  - Why they're difficult to use
- When a mesh isn't good enough
  - Problem areas besides surface models
- A simple manifold
  - Sphere, torus, plane, etc.
- Using the manifold for parameterization
  - Fitting to existing meshes
  - *Basic tools

# What is a manifold?

- Surface analysis tool
  - Developed in the 1880’s
  - Describe surface as a collection of overlapping disks
    - Infinite number
  - Atlas
    - Pages are disks
    - World is surface

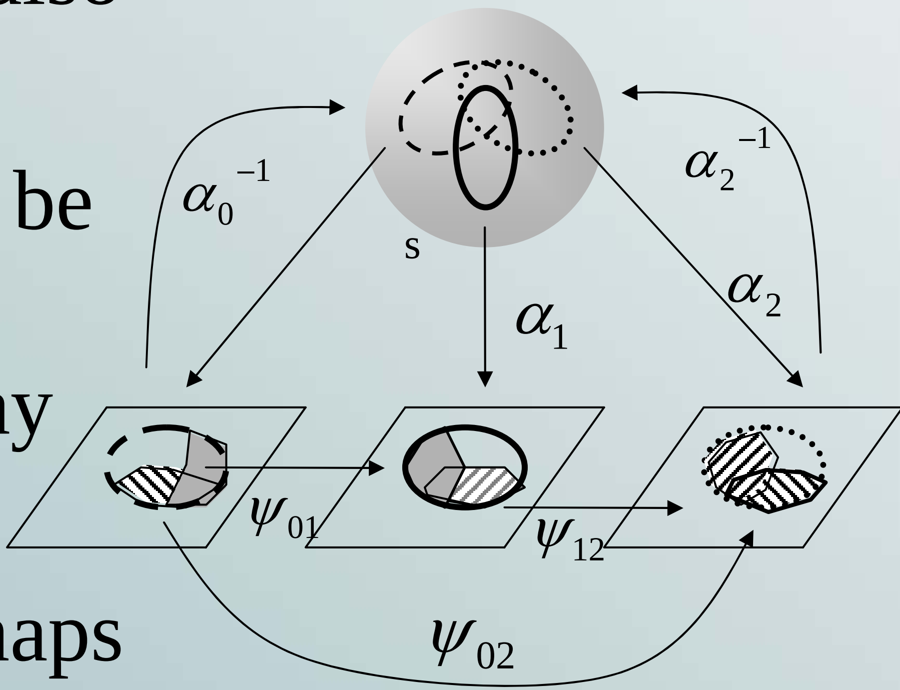


# Formal definition

- Given: Surface S of dimension m embedded in $\Re^n$
- Construct a set of charts, each of which maps a region of S to a disk in $\Re^m$
  - Mapping must be 1-1, onto, continuous
  - Every point in S must be in the domain of at least one chart
  - Collection of charts is called an atlas
- Note: A surface is manifold if such an atlas can be constructed

# Additional definitions

- Co-domain of chart can also be called a chart
- Define the overlap $U_{ij}$ to be the part of chart i that overlaps with chart j. May be empty.
- Transition function $\psi_{ij}$ maps from $U_{ij}$ to $U_{ji}$.



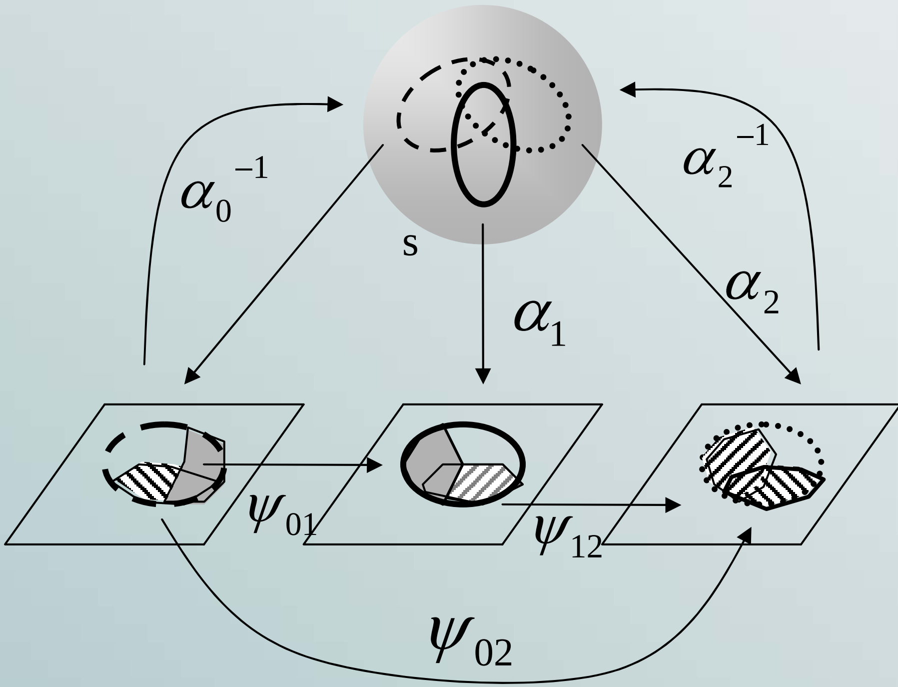

Cindy Grimm

# Going the other way

- Given a set of charts and transition function, define manifold to be quotient
  - Transition functions
    - Reflexive $\psi_{ii}(x) = x$
    - Transitive $(\psi_{ik}(\psi_{kj}(x)) = \psi_{ii}(x)$
    - Symmetric $\psi_{ii}(\psi_{ji}(x)) = x$
  - Quotient: if two points are associated via a transition function, then they're the same point
    - Chart points: chart plus (x,y) point
    - Manifold points: a list of chart points

# Manifold definitions

- Define S, define atlas
  - Overlaps, transition functions secondary
- Define chart domains, overlaps, transition functions
  - Manifold defined by quotient
  - *No geometry*



# Adding geometry

- Define an embedding for each chart
  - $C^k$, e.g., spline patch $E_c : \Re^m \to \Re^n$
- Define a proto-blend function for each chart
  - $C^k$, k derivatives zero by boundary, non-zero on interior, e.g., spline basis function $\hat{B}_c : \Re^m \to \Re$
- Define chart blend function
  - Zero outside of chart $B_c : S \to \Re$
  - Partition of unity

$$B_c(p) = \frac{\hat{B}_c(\alpha_c(p))}{\sum \hat{B}_{c'}(\alpha_{c'}(p))}$$

# Adding geometry, cont.

Media and Machines

- Final surface is a blended sum of chart embeddings

$$E : S \to \Re^n$$

$$E(p) = \sum B_c(\alpha_c(p)) E_c(\alpha_c(p))$$

# Previous work

Media and Machines

- Grimm '95
  - Mesh: one chart for each vertex, edge, face
    - N different transition function types
    - Vertices of valence 4
- Navau and Garcia, 2000
  - Planar mesh: Map to plane
  - General mesh: Use subdivision to separate extraordinary vertices
    - Specific flattening of extraordinary vertices into plane
- Lewis and Hughes '96
  - Complex plane, "unwrap" faces around vertex

# Why is this hard?

- Finding charts, transition functions that are correct is hard
  - Start with mesh
    - Combinatorial or number of charts explosion
    - No linear set of functions (?)
  - Start with points
    - Analytic function?

# Why not just a mesh?

- Surface modeling: smooth, locally parameterized, analytical
- Functions on meshes
  - Texture synthesis, reaction-diffusion
  - Curvature calculation
  - Visualization of data on surface
    - Fluid flow: 2D vector drawing routines
- Manifolds have in-built notion of local neighborhood, moving along surface

# A simple manifold

- Define one manifold for each genus
  – Push geometric complexity into embedding
  – Charts simple (unit square), few in number
  – Substantial chart overlap
  – Transition functions simple
    - Define by mapping to and from a canonical surface (plane, sphere, torus, etc.)
- Simplifies defining functions on manifold

# Roadmap

- Manifold definition for sphere, plane, torus, n-holed torus
- Embedding functions for manifold
- “Fitting” a manifold to an existing mesh
  - 1-1, onto mapping from mesh to manifold

# Plane manifold

- One chart (unit square)
- One transition function (identity)

# Sphere manifold

- S is the unit sphere
- 6 charts, one for each direction
  - Mapping functions are variations of
    - ( $\cos(\theta)\cos(\phi),\sin(\theta)\cos(\phi), \sin(\phi)$ )
  - Inverse functions are found by arcsin, arctan
    - $\phi = \text{asin}(z)$, $\theta = \text{atan2}(y,z)$

# Sphere manifold

Media and Machines

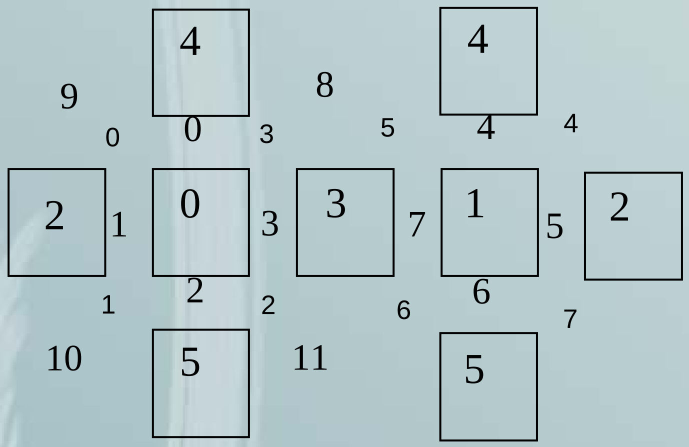
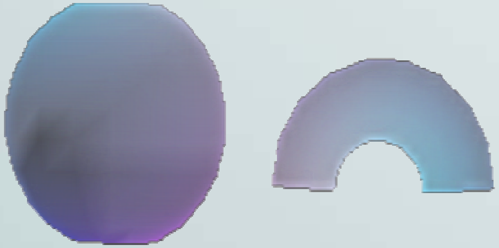
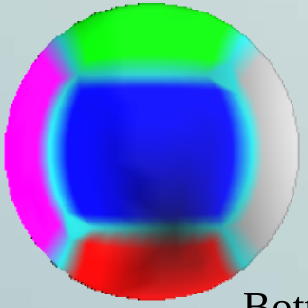



**Chart (squares), edge, and corner indices**



**A single chart on the sphere viewed from the side and top**

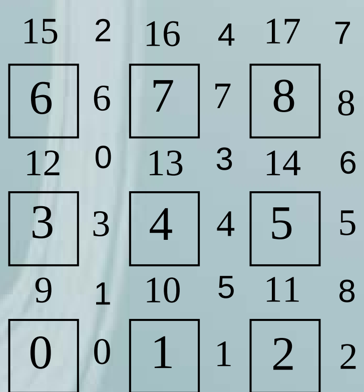
Top cap

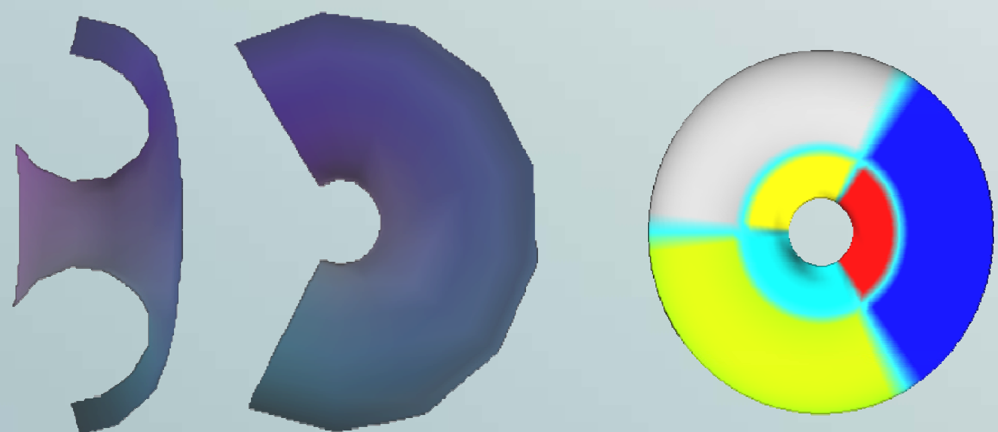


Bottom cap

# Torus manifold

- S is torus of inner radius 0.25, outer 1.25
  - $S(\theta,\phi)=( 1.5 + \cos(\theta)\cos(\phi), 1.5 + \sin(\theta)\cos(\phi), \sin(\theta) )$
  - Domain 0,2pi X 0,2pi
- 9 charts, each shifted and scaled portion of 2pi domain

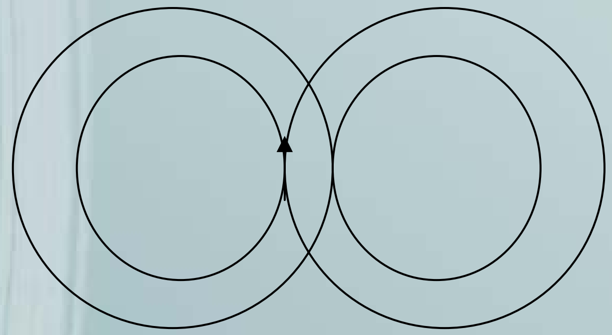


Torus chart

# N-holed torus

- N-copies of torus, with one edge identified in opposite direction



# Embeddings

- Hierarchical splines
  - Spline surfaces have different sized non-zero domains

$$E_c(x,y) = \sum s_i(x,y)$$

- Radial basis functions

$$E_c(u,v) = \left((x_0 + x_1 u + x_2 v) + \sum w_i \phi(\| (u,v) - c_i \|)\right)_{x,y,z}$$

# Tessellation

- Triangulate interior of each chart, stitch together along boundary edges
- Further split faces if needed
  - Area taken up in final embedding

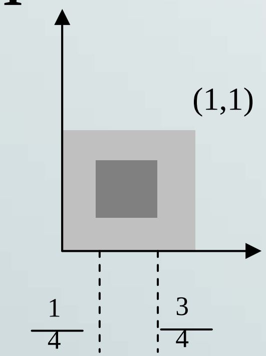
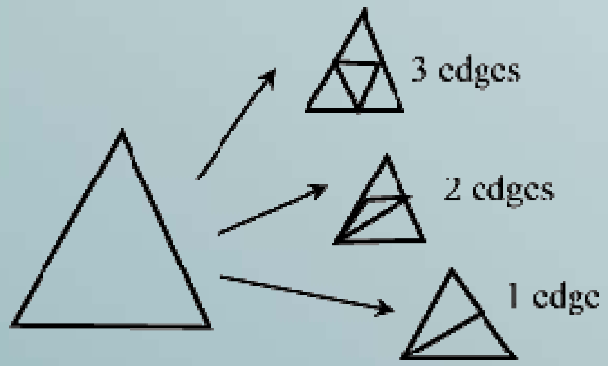
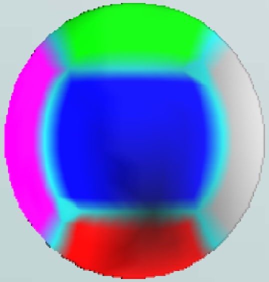



Chart interior



# Mapping from mesh to manifold

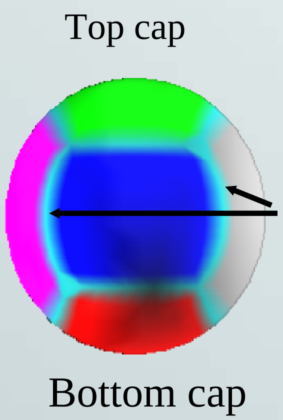
- 1-1, onto function
- Map vertices first
- Map faces using barycentric coords
  - Requires that there exist a chart such that all three vertices of face map into that chart
  - Plane, torus : usual barycentric coords
  - Sphere: spherical barycentric coords

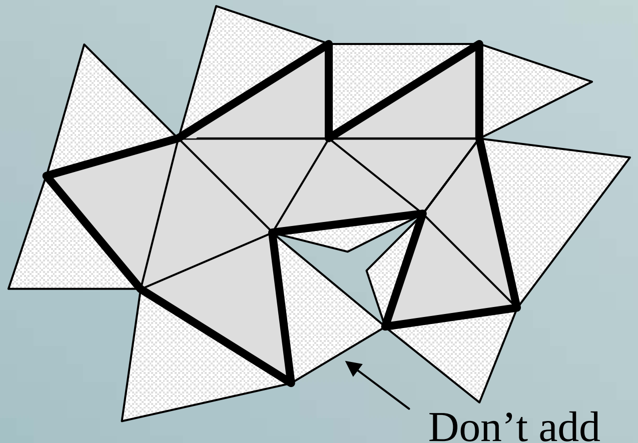
# Mapping vertices

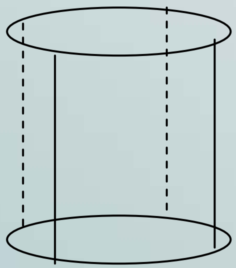
- Graph partition problem
- Divide mesh into n regions (for n charts) that meet with correct topology
  - Project region into interior of chart
- Criteria:
  - No folding (star of vertex forms convex polygon in chart)
  - Roughly same number of faces in each chart
  - Each face has one chart it maps into

# Sphere algorithm

- Grow top cap (disk)
- Grow bottom cap
- Join boundary with four edges
  - Shortest path
  - Vertices must be accessible

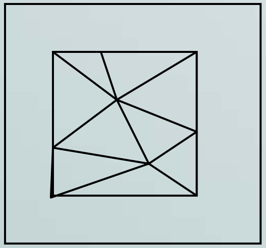


# Sphere algorithm, cont.

- Project each region onto chart
- Run Floater's algorithm to place vertices in chart
  - Least squares problem
  - Boundary points are fixed
    - $Loc(v_i) = (u,v)$
  - Place interior vertices at centroid of vertex star

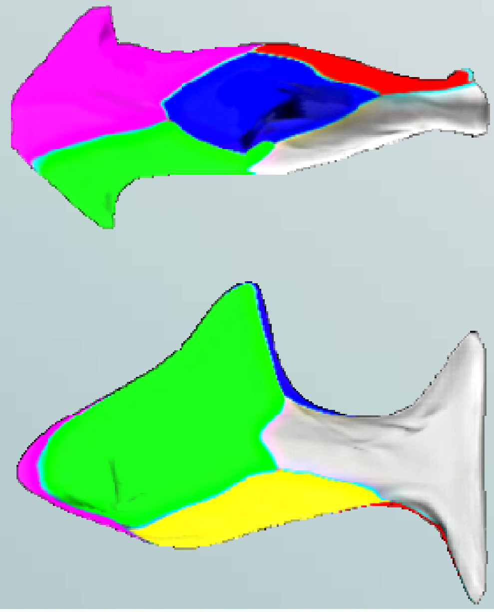


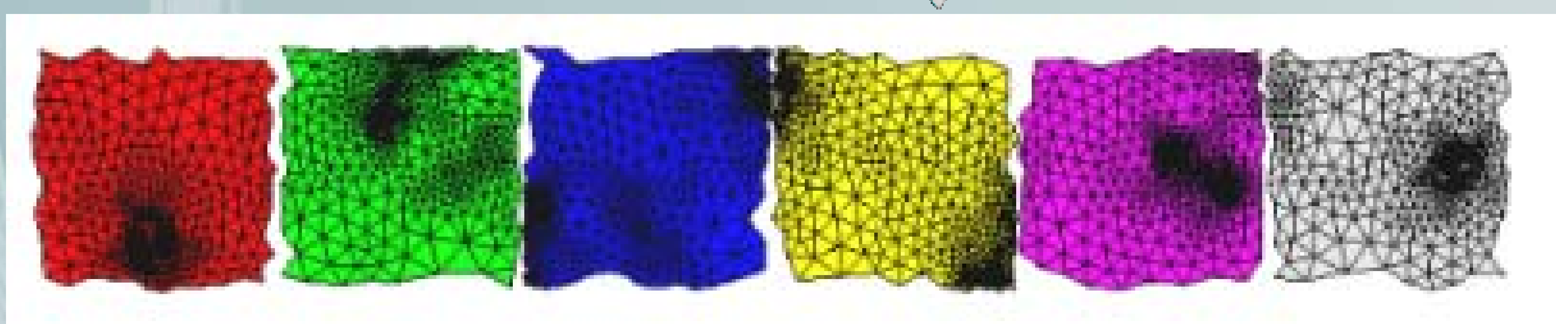
$$v_i - \sum_j v_j = 0$$

# Sphere algorithm, cont.

- Adjust projection until criteria met
  - Get disk of faces in chart
    - Grow (or shrink) placement based on percentage of faces
    - Reproject
  - Move vertices towards centroids

# Results



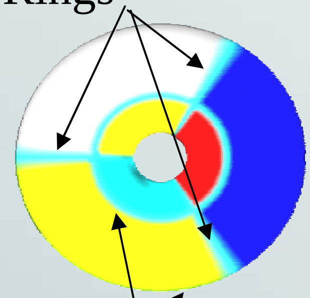


# Plane algorithm

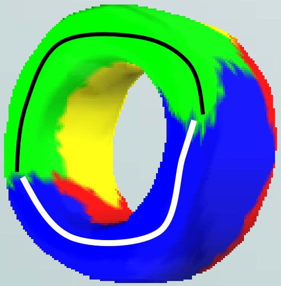
- Map boundary vertices to boundary of chart
  - Interior vertices mapped to centroid of vertex star
  - Floater’s algorithm

# Torus algorithm

- Want three rings in 2 directions
  - Grow four disks, seeded far apart
  - One half of loop goes through one disk, other half through adjacent disk
    - Disks meet in two disjoint regions
  - Grow two annuli out from disk
    - Parallel loop
    - Grow one disk faster
  - Repeat



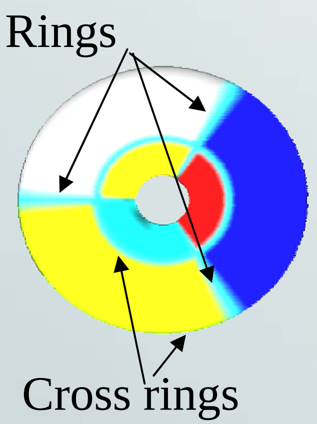

Four disks, path through green disk plus path through blue disk makes loop

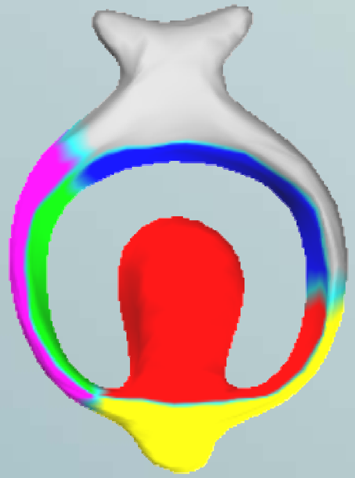


# Torus algorithm, cont.



- Three paths between loops
  - Total of 9 shortest paths
- Given 9 initial regions, adjust



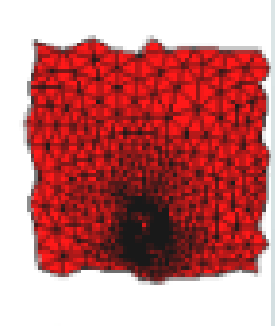



# Algorithm analysis

- Best suited for 500-10,000 vertices
- Not guaranteed to reach solution
  - “narrow” spots, not enough vertices around tube
    - Use subdivision to produce more vertices
  - Solution exists
- Alternative approaches
  - Use progressive meshes to simplify to base case
    - Initial vertex mapping
  - Add vertices back in

# Fitting embeddings

- Embedding function can be arbitrarily complex
  - Not uniform
- Radial basis functions
  - Uneven distribution is not a problem
  - Function complexity grows with number of points
- Hierarchical splines
  - Need to find best patch placement



# Fitting radial basis functions

- One equation each for x,y,z
- N vertices mapped to chart implies N basis functions
- Set coefficients so surface passes through N points
  - Linear equations

$$E_c(u,v) = \left((x_0 + x_1 u + x_2 v) + \sum w_i \phi(\| (u,v) - c_i \|)\right)_{x,y,z}$$

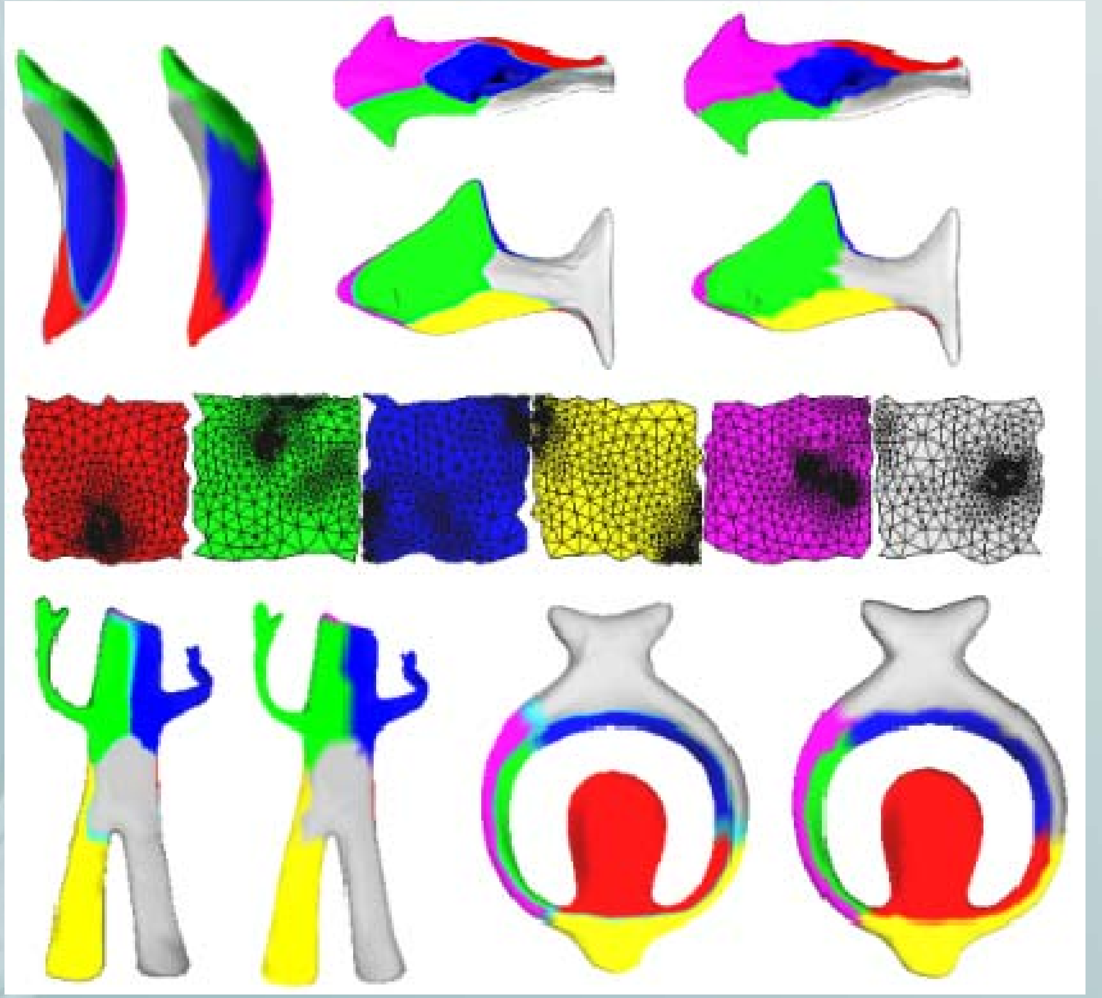
# Fitting hierarchical splines

- Error metric
  - Add patch where error is bad
  - Distance from surface to mesh
    - 1-1 onto mapping
- Grid domain and evaluate at grid squares
  - Max(error at square, vertices in square)
  - Add patch over largest contiguous set of bad squares
    - Max and average thresholds

# Hierarchical splines, cont.

- Add patches until error metric below threshold
  - Find  contiguous bad squares
  - Fit patch to remaining error
    - Domain based on bad squares
    - Fit to evenly spaced points in domain
- Refit with all patches simultaneously

# Results



Cindy Grimm

Washington University in St. Louis
School of Engineering & Applied Science

# Summary so far

- Establish 1-1, onto correspondence between mesh and manifold
- Can construct embedding to create smooth surface
  - Approximate mesh (splines)
  - Exact fit (radial basis functions)

# Basic tools (in progress)

- Functions on manifold
  - Ratio of parameter area to surface area
    - Evenly distribute points on surface
  - Vector field in parameter space
    - “up”, fluid flow, user-defined, principle curvature
    - Continuous as possible
      - Evenly spaced stripes
  - BRDF

# Basic tools (in progress)

- Surface traversal
  - Given a set of points, produce an ordering
    - Texture synthesis
      - Upper left points already visited
    - Spiral
    - Along flow paths
- Texture mapping
  - Uniform/non-uniform density of pixels
  - Original mesh

# Conclusion



- Manifolds have potential as parameterization tool
  - Meshes, implicits, patch collections
  - Function display on surface, non-photorealistic renderings, growing plants on surfaces
- Simple manifolds with easy-to-use tools
  - Computationally simple

# Other future work

Media and Machines

- Surface modeling interface
- Link between subdivision surfaces and manifolds
- N-holed torus mapping to mesh