Meshing: A (Biased) Crash Course

Mathieu Desbrun

The Big Picture

Meshing, an essential preprocessing step

- For surface representation
  - complex geometry
  - with a few basic geometric primitives

- For simulation of physical phenomena
  - realistic/accurate animation of fluid, deformable solids, electromagnetism
    - often modeled as PDE
  - domain discretization
    - Then FEM/FVM to integrate PDE in space (& time)

What is Meshing?

General Idea: "flat" 2D "flat" 3D

- breaking up a physical domain
  - 2D domain in 2D, or in 3D, or 3D domain...
  - into simpler subdomains—the elements
    - simplices (triangles, tetrahedra)
    - or not (quads, polygons, hexahedra, polyhedra)

- Then FEM/FVM to integrate PDE in space (& time)
Brief Glossary

Node, element:

Structured/Unstructured Mesh:
- regular valence and degree

Isotropic/Anisotropic Mesh:
- without/with stretched elements

Graded Mesh:
- w/ elements varying in size (fct of position)

Why (Un)Structured?

Structure brings:
- simpler data structure
- better compression
- reuse of DSP algos
  - smoothing
  - wavelets

Why Anisotropy?

Isotropic meshing

Anisotropic meshing

Brief Glossary

Node, element:

Structured/Unstructured Mesh:
- regular valence and degree

Isotropic/Anisotropic Mesh:
- without/with stretched elements

Graded Mesh:
- w/ elements varying in size (fct of position)
Why Gradation?

Allows for better capture of details for same vertex budget

Goals in Mesh Generation

- Control over shape of elements
  - mass matrix conditioning
  - shape often induced by PDE
    - good bet: regular shape
- Control over sizing
  - often dictated by simulation ... & boundary
- Control over total size
  - low number of elements is preferred

How to Mesh a Domain?

Usually, ‘bottom-up’ approach
  - vertices are first placed (point sampling)
  - then connected by edges, triangles, etc...
But not always...
  - remeshing thru simplification (top-down)
  - placement of edges first [Alliez et al. 2003]
    - as strokes along important directions
  - placement of elements first [Alliez et al. 2004]
    - to best approximate local shape, based on normal field

How to Mesh a Domain?

Usually, ‘bottom-up’ approach
  - vertices are first placed (point sampling)
  - then connected by edges, triangles, etc...
But not always...
  - remeshing thru simplification (top-down)
  - placement of edges first [Alliez et al. 2003]
    - as strokes along important directions
  - placement of elements first [Alliez et al. 2004]
    - to best approximate local shape, based on normal field

What We Won’t Mention [Much]

Meshing of curved manifolds
  - often, can be flattened thru parameterization
    - see STAR report aim@shape, Alliez et al. 2004
Meshing of non manifold objects
Polygonal Meshing
Surface Approximation
  - vertices can be off the original surface
    - see [Nadler 98, Simpson ’94, Heckbert-Garland ’99]
So How do I Mesh a Domain?

Seemingly, vertex placement is crucial
- we must “sample” the domain appropriately
- well-spaced point sets?
  - spread nodes in the (flat or curved) domain
  - spread them evenly
    - using an attraction/repulsion simulation for instance
- done, right?
  - most papers in graphics use this approach
- good news: sufficient in 2D

Introducing Delaunay Triang.

‘Canonical’ triangulation of a point set
- with numerous optimal properties
  - see: Eppstein, Meshing Roundtable 2003
  - Shewchuck, his UC Berkeley’s website
  - Lots of robust, existing codes (Triangle, CGAL, …)
- “territories” = Voronoi diagram
- “dual” = Delaunay triangulation (DT)
- Live demo

Moving on to 3D

Everything breaks...
- the regular tet does NOT tile space
  - perfectly shaped tets everywhere not attainable
- DT based on well-spaced points have slivers

Scrutinizing the Beast

Four (well-spaced) vertices near the equatorial plane of their circumsphere

Taming the Beast?

Recipes to get rid of slivers
- local jiggling of faulty vertices
  - through optimization of shape’s “quality”
    - for example, penalize small dihedral angles
  - often, non-linear and slow
  - no guarantee of success—in fact, may be worse
- locally altering the Delaunay triangulation
  - called “sliver exudation”
  - no longer Delaunay, obviously
  - guarantees, but no practical bounds
### Popular 3D Meshing Methods

<table>
<thead>
<tr>
<th>Delaunay-based</th>
<th>spring energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>refinement</td>
<td>Laplacian (bad!)</td>
</tr>
<tr>
<td>sphere packing</td>
<td>non-zero rest length</td>
</tr>
<tr>
<td>specific subdivision</td>
<td>aspect / radius ratios</td>
</tr>
<tr>
<td>octree</td>
<td>dihedral / solid angles</td>
</tr>
<tr>
<td>regular lattice</td>
<td>max-min/min-max</td>
</tr>
<tr>
<td>advancing front</td>
<td>volumes, lengths</td>
</tr>
<tr>
<td></td>
<td>sphere radii</td>
</tr>
<tr>
<td></td>
<td>sliver exudation</td>
</tr>
</tbody>
</table>

### An Interesting Alternative

Emergence of variational approaches
- Centroidal Voronoi Tessellation
- Optimal Delaunay Triangulation

Idea:
- position of vertices also *optimal* in some way
- often based on lifting transformation

---

### Lifting Transformation (nD)

Lift points \((x, y)\) to paraboloid \((x, y, x^2 + y^2)\)
- placement of origin irrelevant

Delaunay triang. = convex hull of 3D points

### Lloyd Algorithm in 3D

Breaks too, unfortunately...
- why? dual cells well-shaped
- but not sufficient for well-shaped tets!
- use of a dual approach, maybe?
Under/Overlaid Approximant

CVT
- approximant
- compact Voronoi cells
- isotropic sampling

Results of Optimization in 2D

Optimization: init

Distribution of radius ratios

Optimization: step 1

Optimization: step 2
Conclusions

- Meshing: a fascinating research field
  - needs theoretical guarantees
  - needs practical results
- Link to DEC
  - improves Hodge star if mesh is nice

Please, consider using Delaunay
- not slow (unlike what you've heard)