

IFT 6113

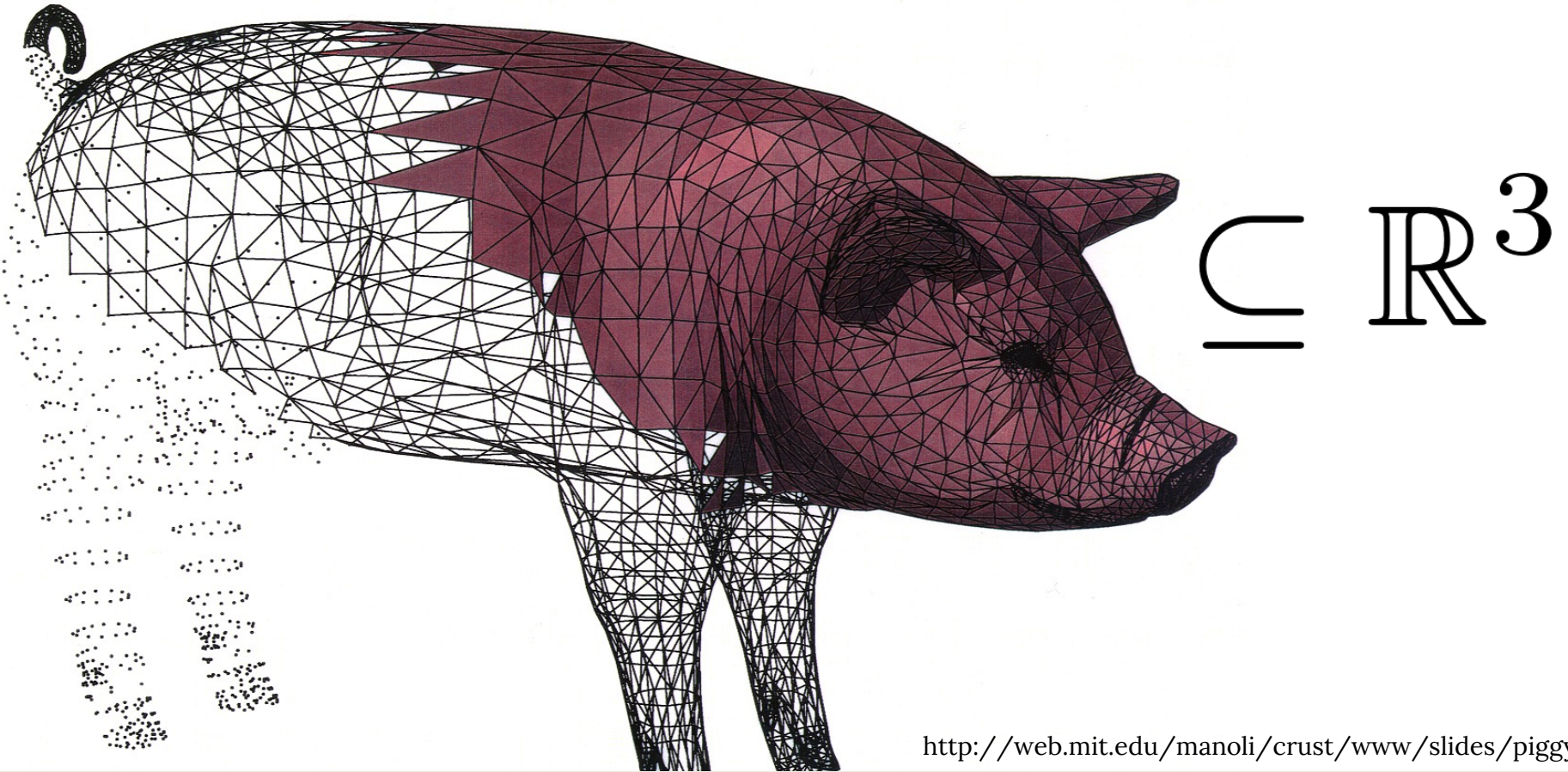
SHAPE REPRESENTATIONS

<http://tiny.cc/ift6113>



Mikhail Bessmeltsev

OUR FOCUS

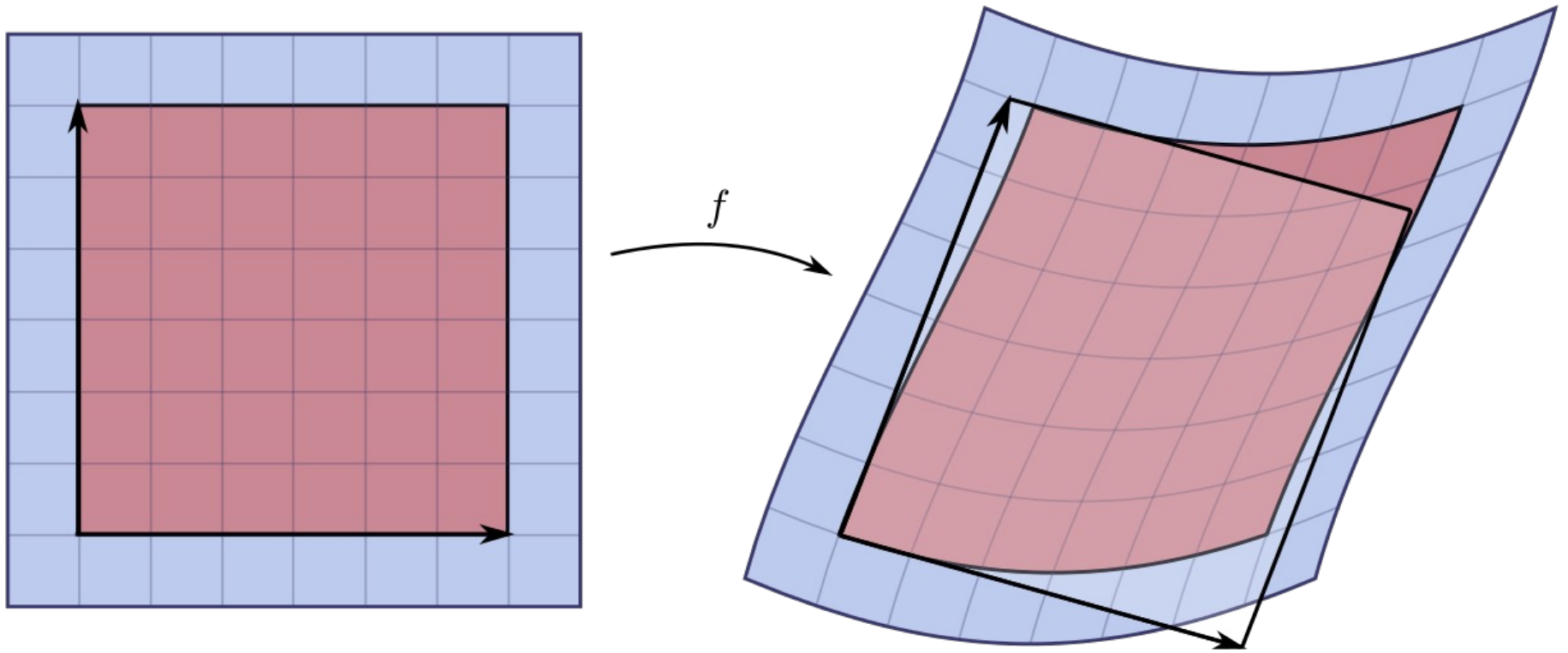


<http://web.mit.edu/manoli/crust/www/slides/piggy.jpg>

Embedded geometry

***What is an
embedded surface?***

PARAMETRIC SURFACE



NEXT ISSUE

$$f : \mathbb{R}^2 \rightarrow \mathbb{R}^3 \dots ?$$

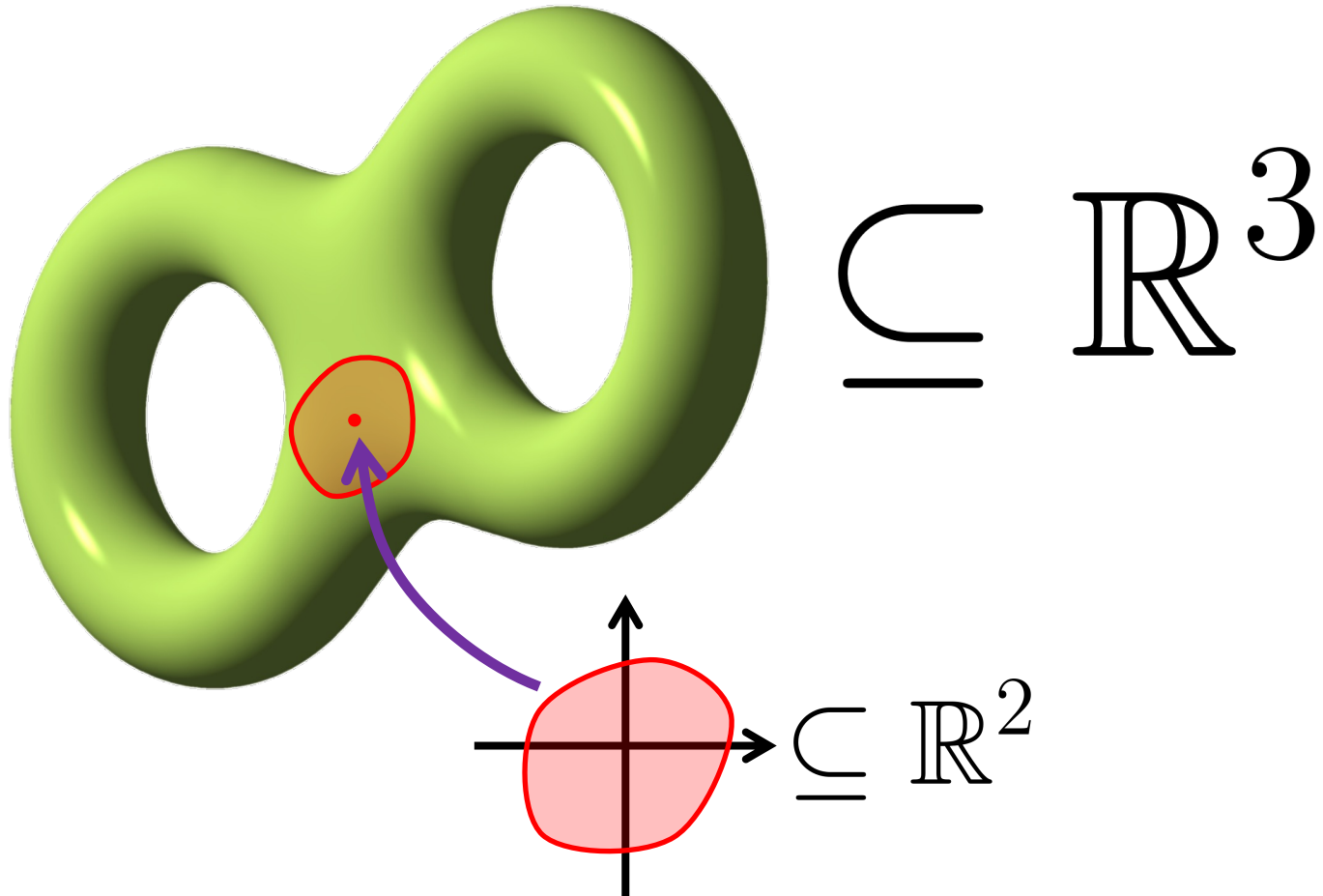


**One function
isn't enough!**

*A surface is a
set of points
with certain properties.*

It is not a function.

THEORETICAL DEFINITION OF SURFACE

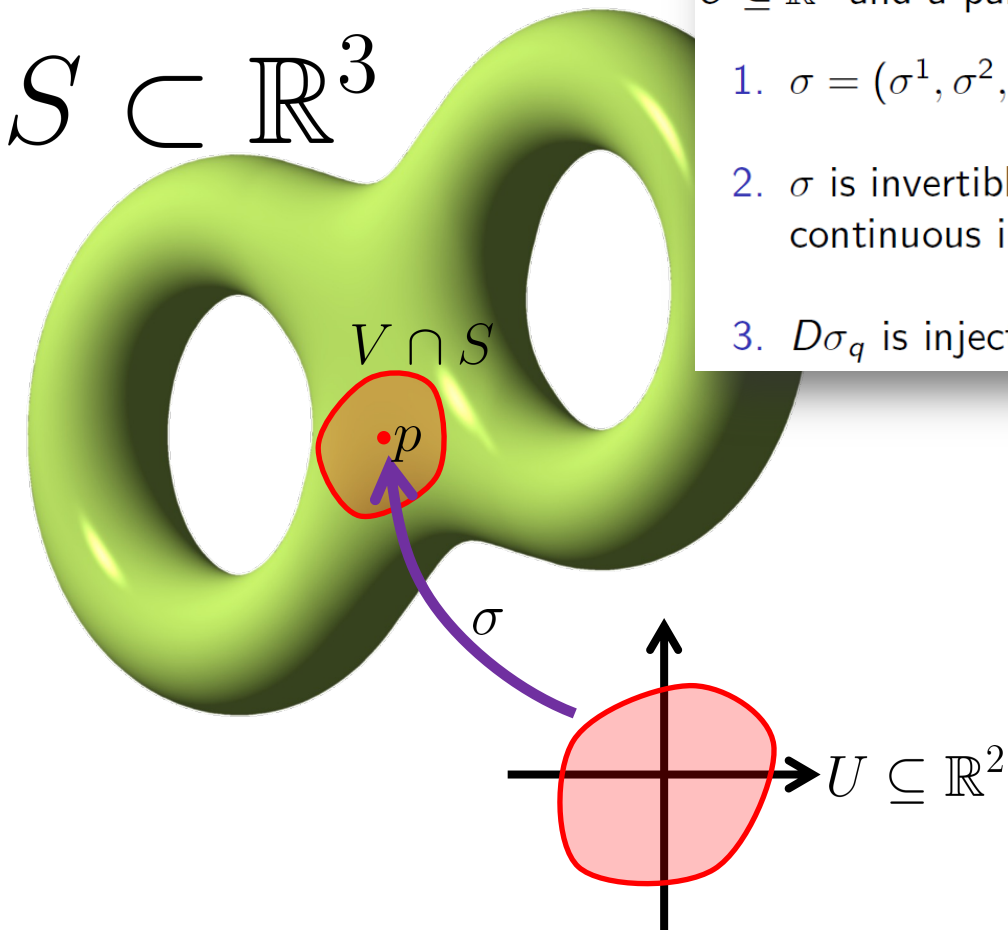


THEORETICAL DEFINITION OF SURFACE

A set $S \subseteq \mathbb{R}^3$ is a *regular surface* if for each $p \in S$ there exists an open neighbourhood $V \subseteq \mathbb{R}^3$ containing p , an open neighbourhood $U \subseteq \mathbb{R}^2$ and a parametrization $\sigma : U \rightarrow V \cap S$ such that:

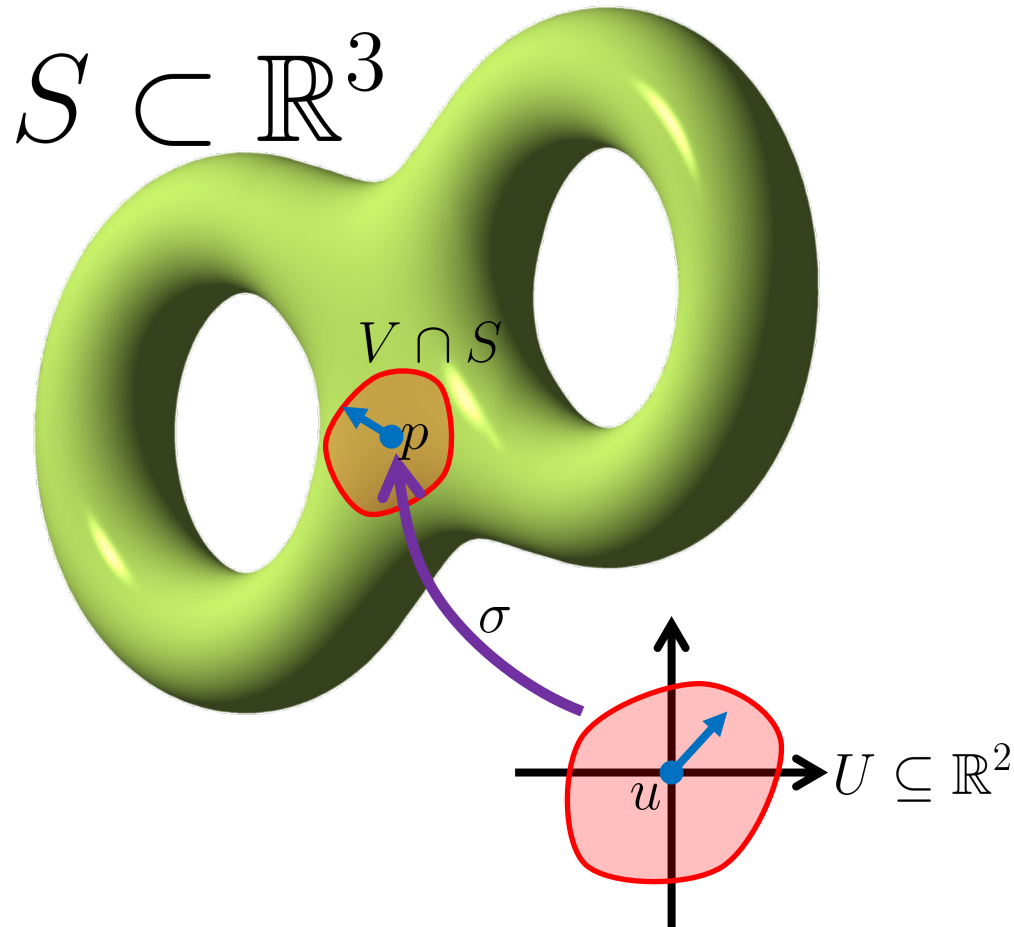
1. $\sigma = (\sigma^1, \sigma^2, \sigma^3)$. is continuous
2. σ is invertible as a map from U onto $V \cap S$ and has a continuous inverse.
3. $D\sigma_q$ is injective $\forall q$. (If and only if $\det((D\sigma_q)^\top D\sigma_q) \neq 0$.)

Text from Stanford CS 468 lecture 5 (2013), A. Butscher



TANGENT SPACE

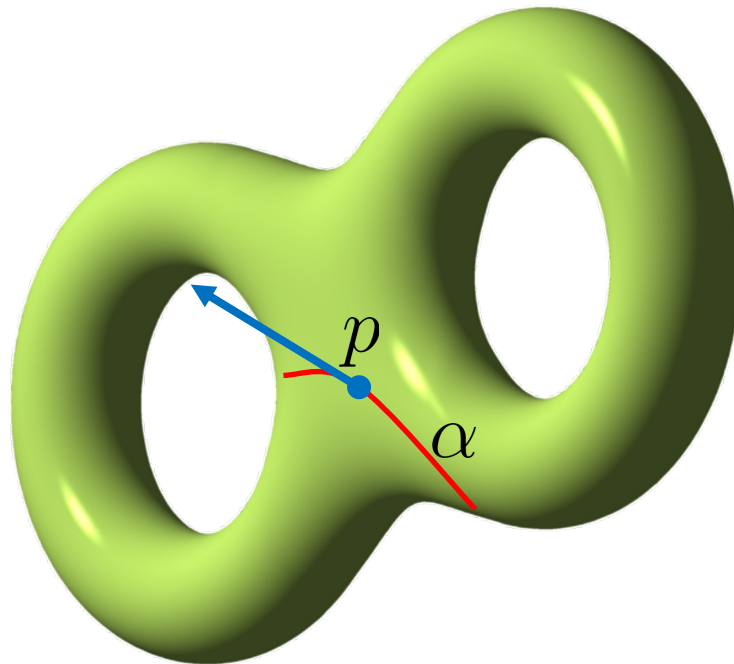
$$T_p S := \text{Image}(D\sigma_u)$$



TANGENT SPACE: COORDINATE-FREE

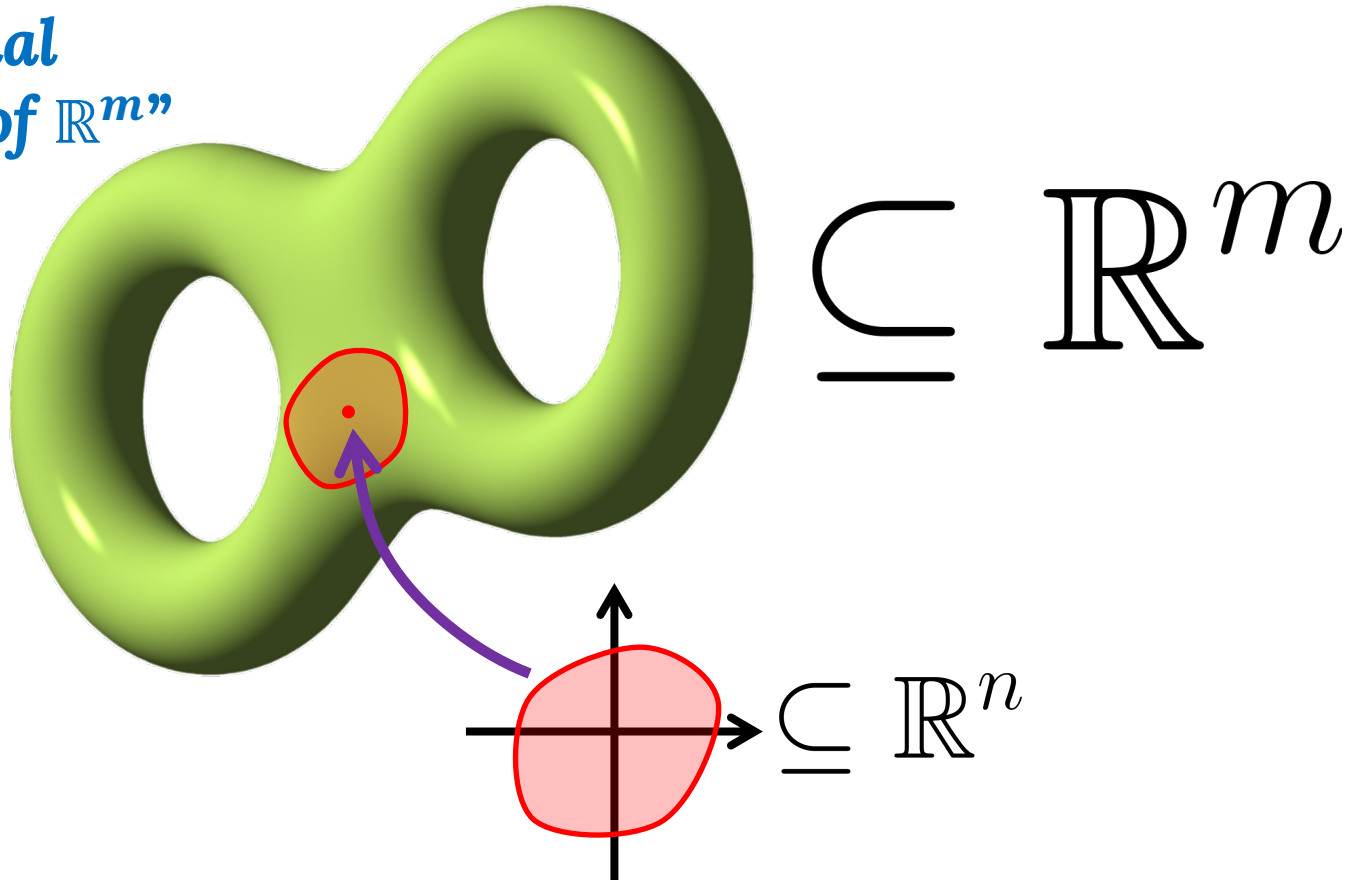
$$v \in T_p S \iff$$

there exists curve $\alpha : (-\varepsilon, \varepsilon) \rightarrow S$
with $\alpha(0) = p, \alpha'(0) = v$



N-D EMBEDDED MANIFOLD

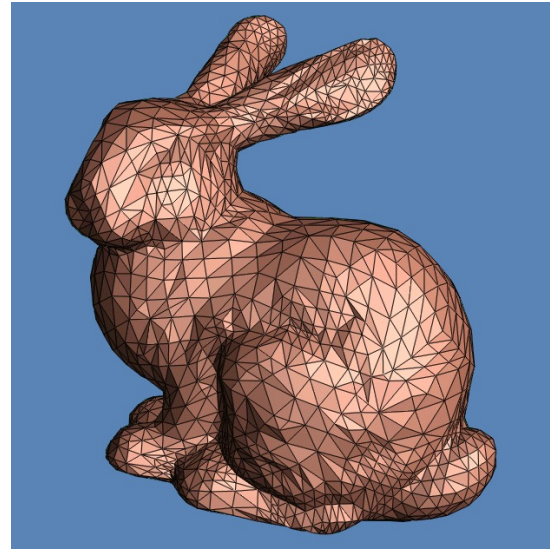
*“n-dimensional
submanifold of \mathbb{R}^m ”*



**Includes curves,
surfaces, 3-manifolds,**

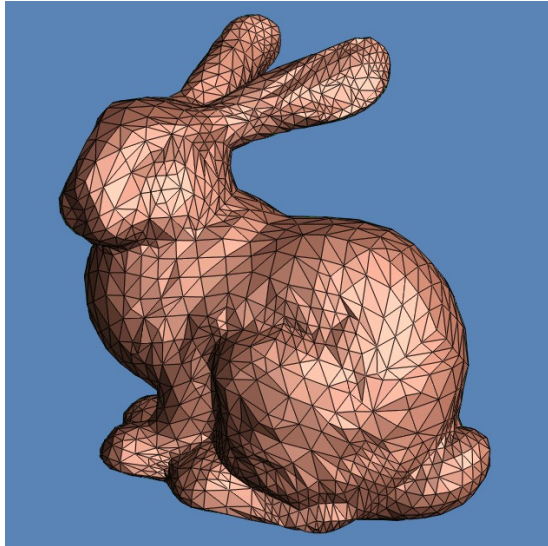
DISCRETE PROBLEM

*What is a discrete surface?
How do you store it?*



SHAPE REPRESENTATIONS

Boundary



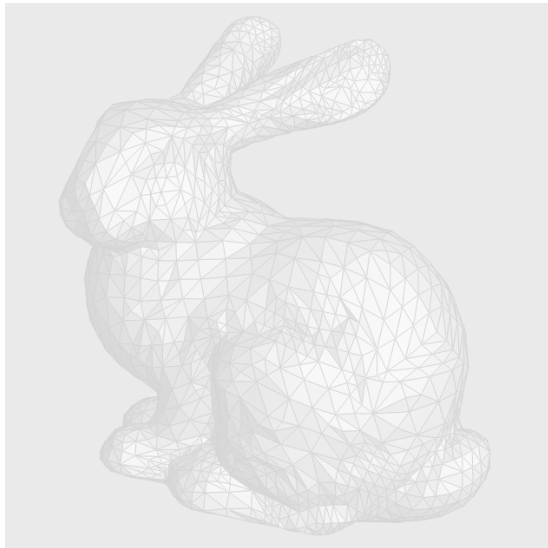
Volume



<https://mathgrrl.com/hacktastic/2015/10/low-voxel-stanford-bunny-voxelizationminecraft-tutorial/>

SHAPE REPRESENTATIONS

Boundary



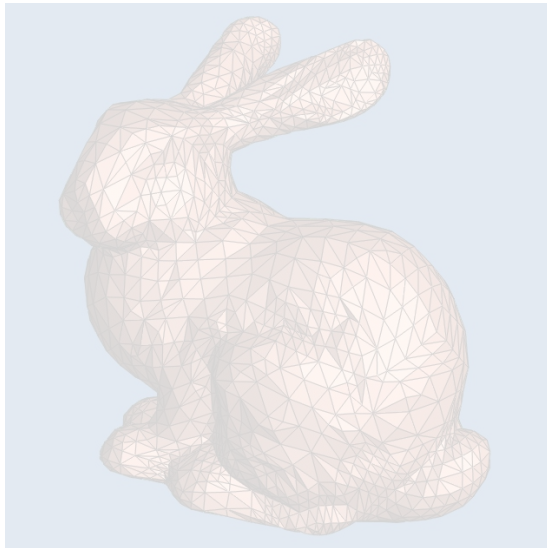
Volume



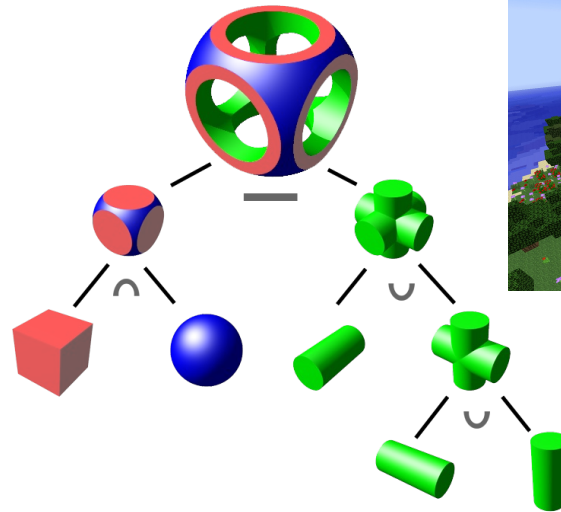
<https://mathgrrl.com/hacktastic/2015/10/low-voxel-stanford-bunny-voxelizationminecraft-tutorial/>

SHAPE REPRESENTATIONS

Boundary



Volume



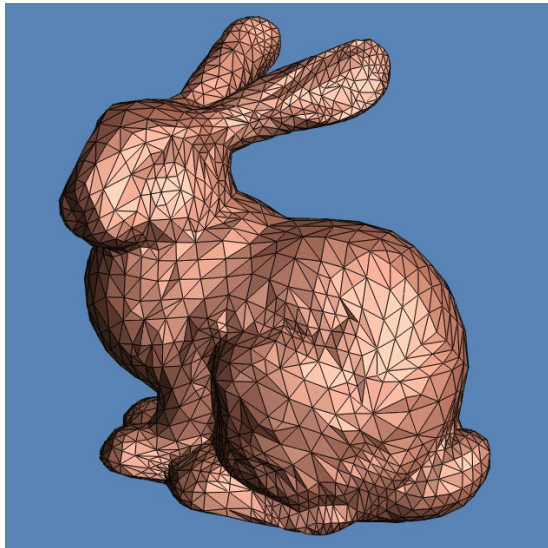
Constructive Solid Geometry



Voxelization

SHAPE REPRESENTATIONS

Boundary

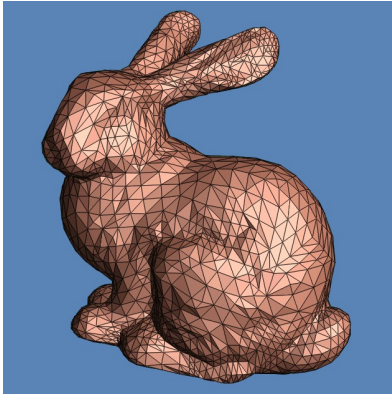


Volume

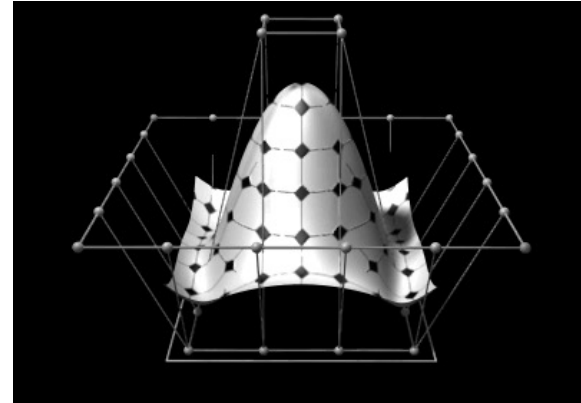


<https://mathgrrl.com/hacktastic/2015/10/low-voxel-stanford-bunny-voxelizationminecraft-tutorial/>

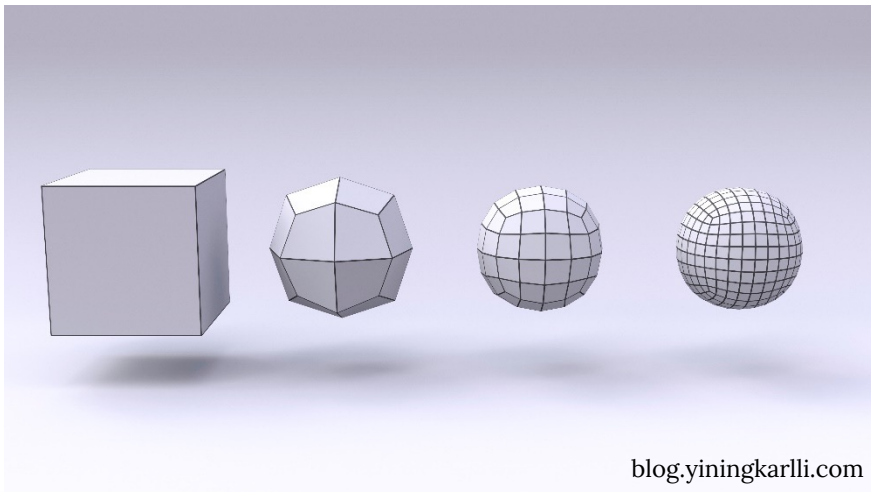
BOUNDARY SHAPE REPRESENTATIONS



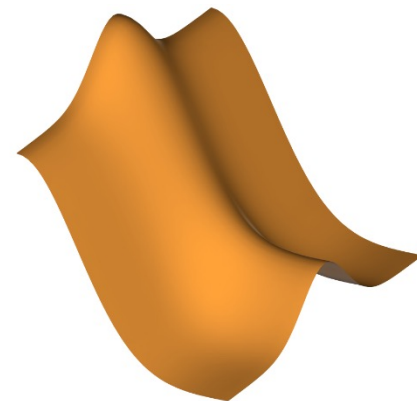
Meshes



NURBS



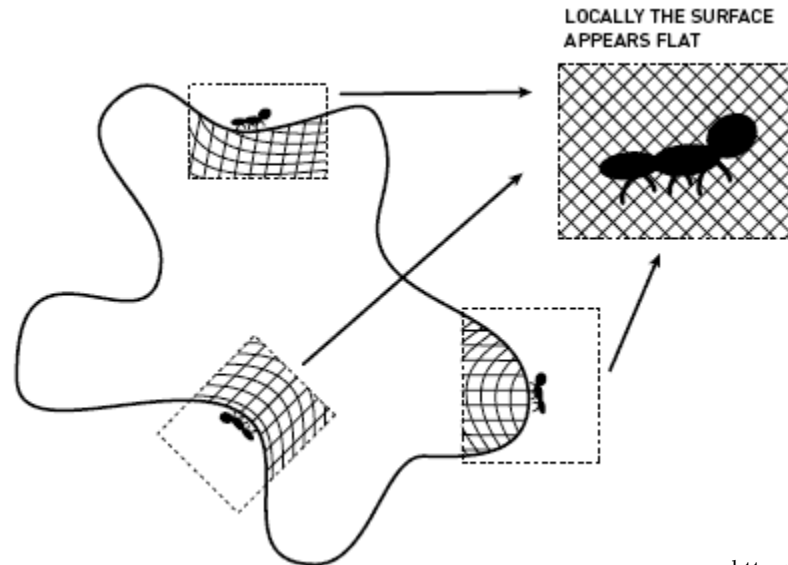
Subdivision surfaces



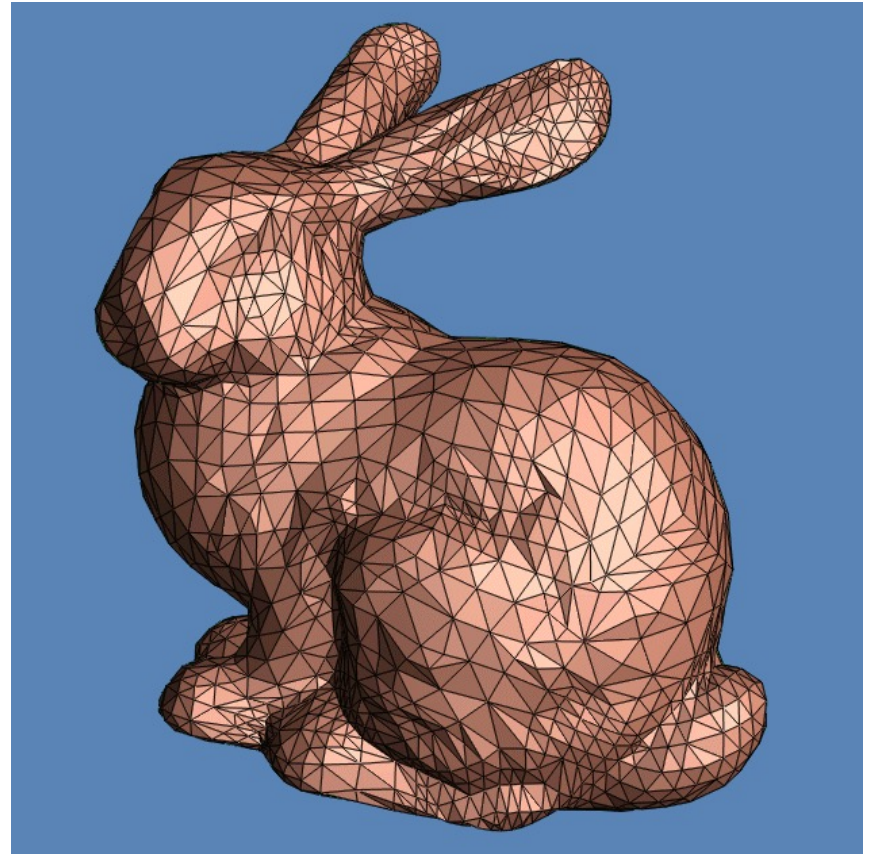
Tensor product surfaces

DIFFERENTIAL GEOMETER'S MANTRA

*A surface is
locally planar.*



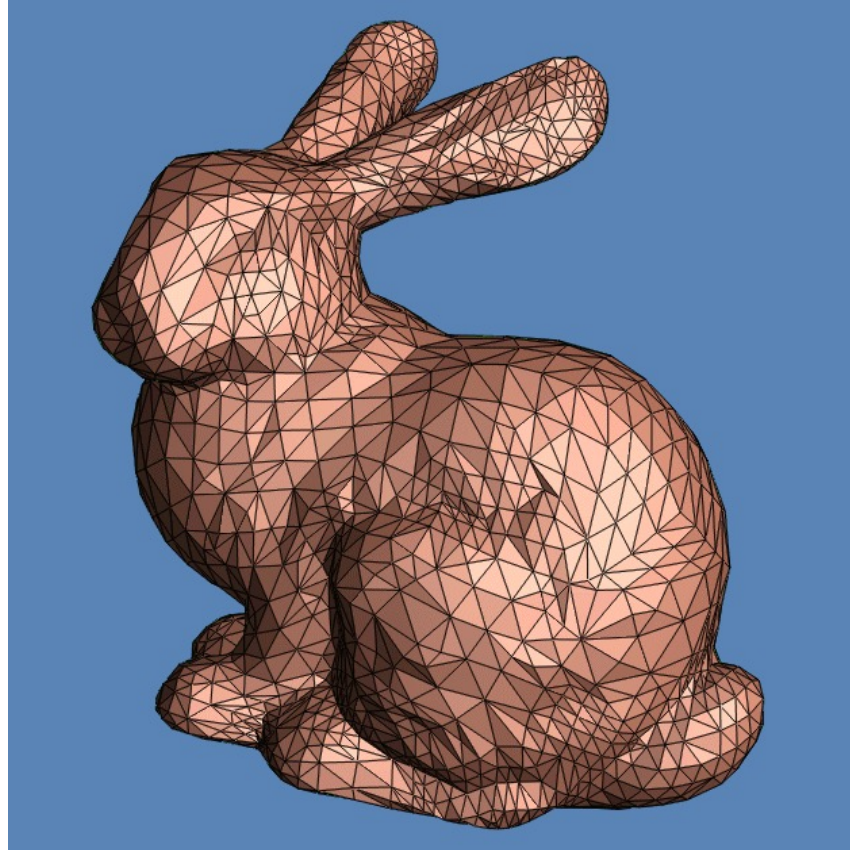
COMMON REPRESENTATION



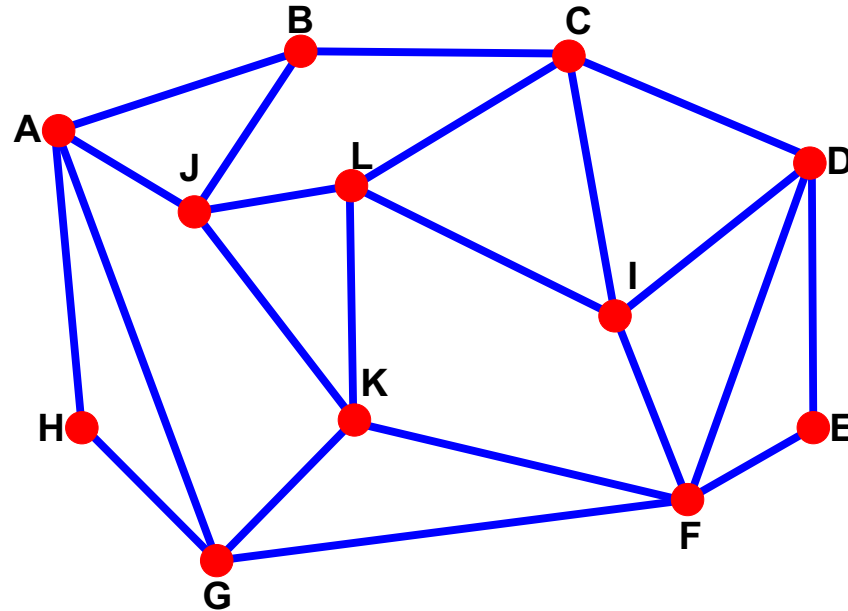
<http://graphics.stanford.edu/data/3Dscanrep/stanford-bunny-cebal-ssh.jpg>
<http://www.stat.washington.edu/wxs/images/BUNMID.gif>

Triangle mesh

MESH = EMBEDDED GRAPH?



GRAPH

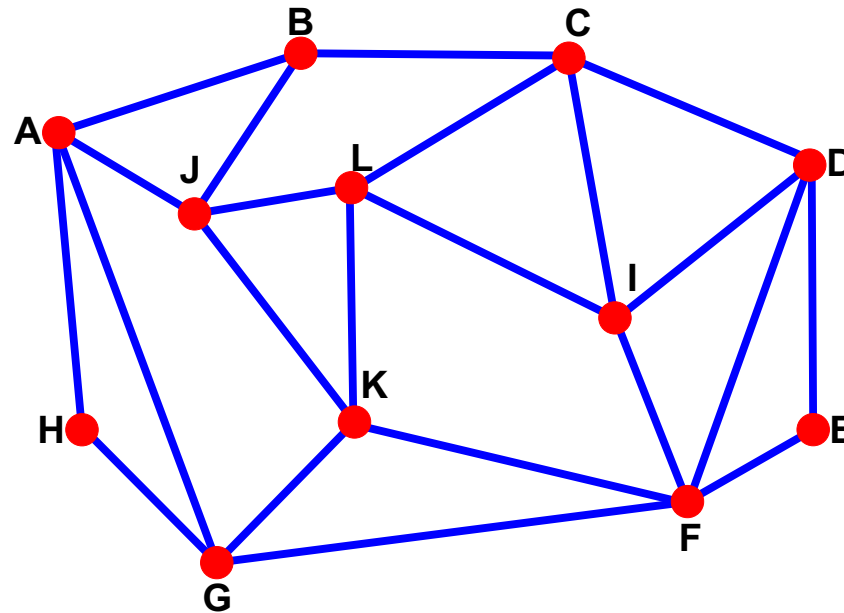


$$G = \langle V, E \rangle$$

$$V = \text{vertices} = \{A, B, C, \dots, L\}$$

$$E = \text{edges} = \{(A, B), (B, C), (C, D), \dots, (L, I)\}$$

GRAPH

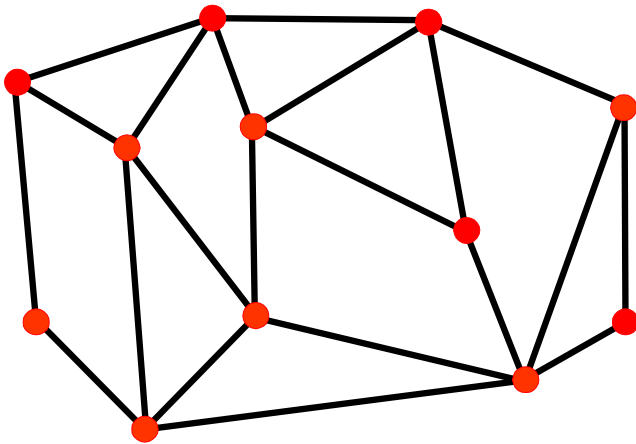


Vertex degree (valence) = number of edges adjacent to vertex

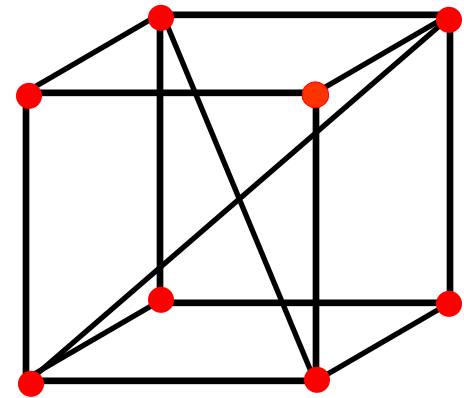
$$\text{deg}(J) = 4, \text{deg}(H) = 2$$

GRAPH EMBEDDING

Graph is ***embedded*** in \mathbb{R}^d
if each vertex is assigned a position in \mathbb{R}^d .

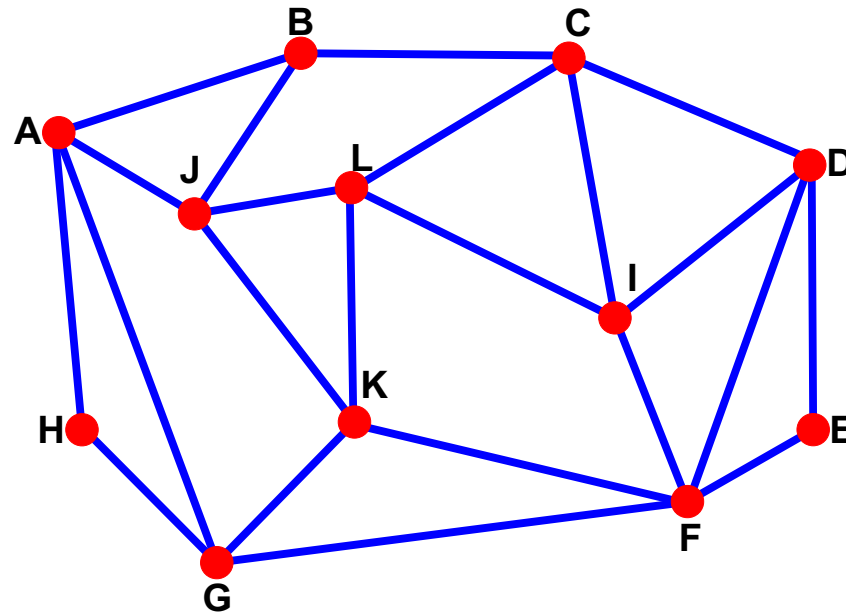


Embedding in \mathbb{R}^2



Embedding in \mathbb{R}^3

GRAPH



Face: cycle of vertices/edges which cannot be shortened

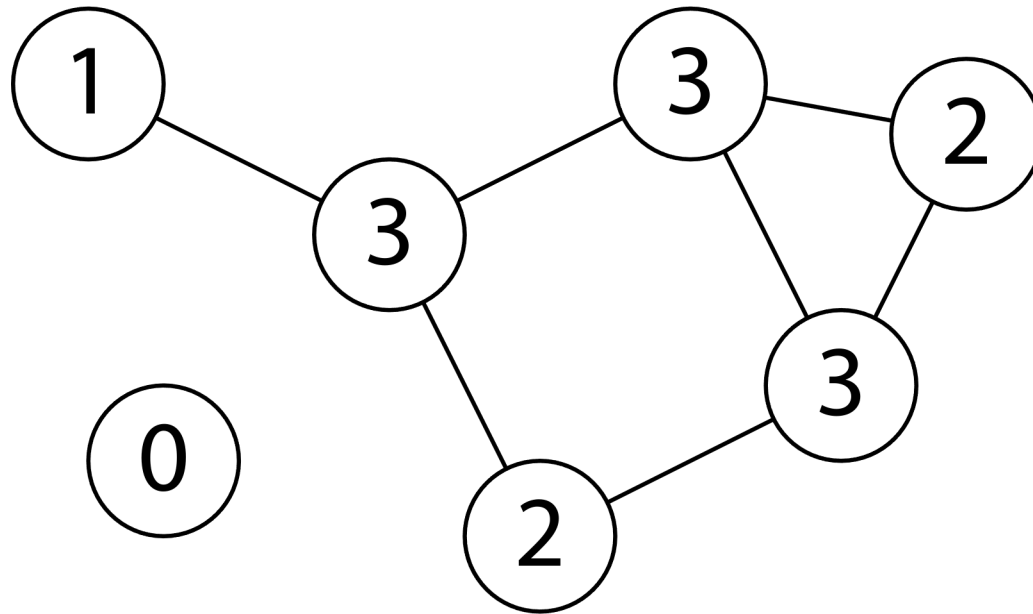
\mathbf{F} = faces = $\{(A,H,G),(A,J,K,G),(B,A,J),(B,C,L,J),(C,I,L),(C,D,I),$
 $(D,E,F),(D,I,F),(L,I,F,K),(L,J,K),(K,F,G)\}$

TRIANGLE MESH

A graph, embedded in 3D,
such that each face is a triangle?

CONNECTED GRAPH

⇔ there is a path of edges connecting every two vertices



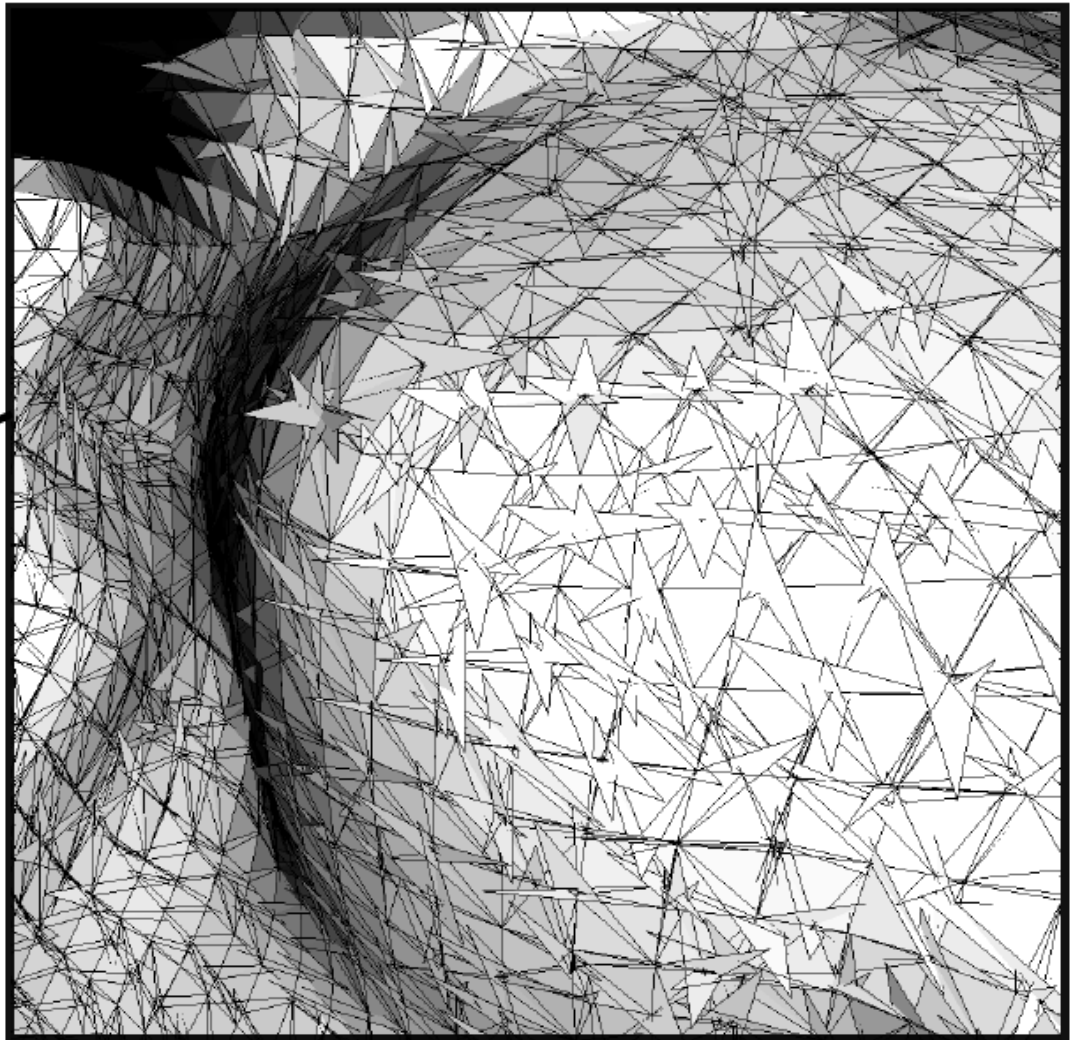
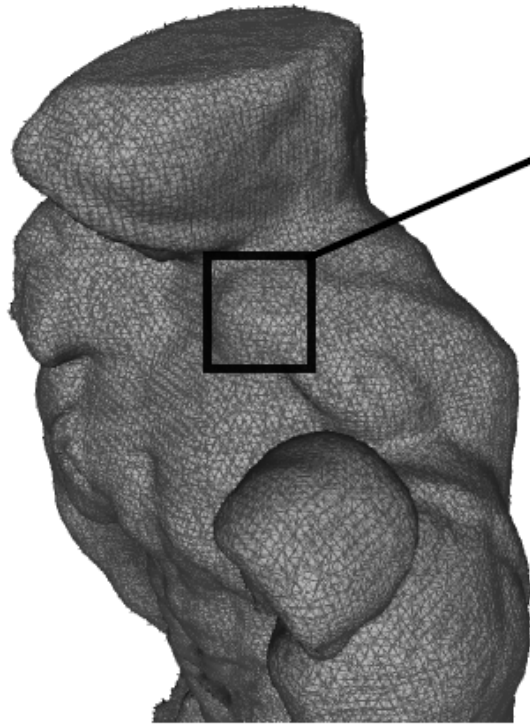
A SIMPLER APPROACH?

$$M = (V, T)$$

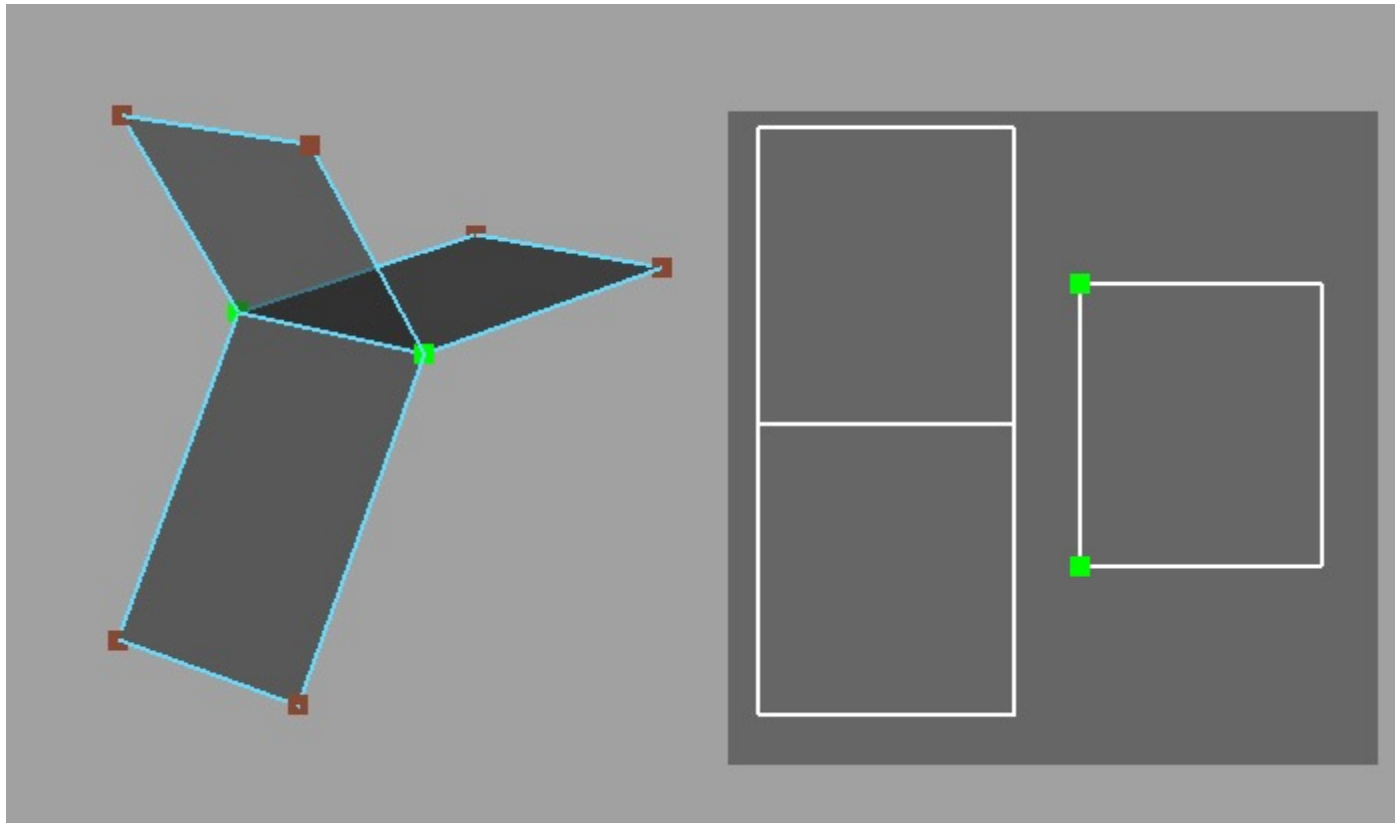
What conditions are needed?

Triangle mesh

IS THIS A SURFACE?

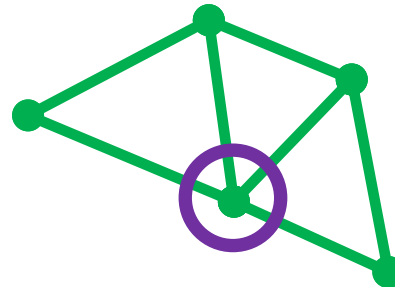
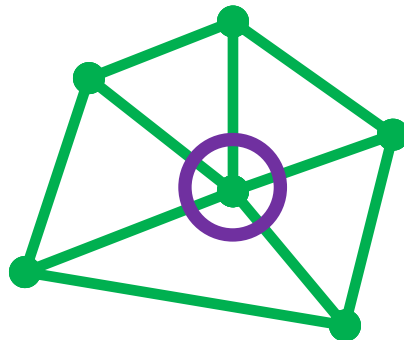


NONMANIFOLD EDGE



MANIFOLD MESH

1. Each **edge** is incident to one or two faces
2. **Faces** incident to a vertex form a closed or open fan



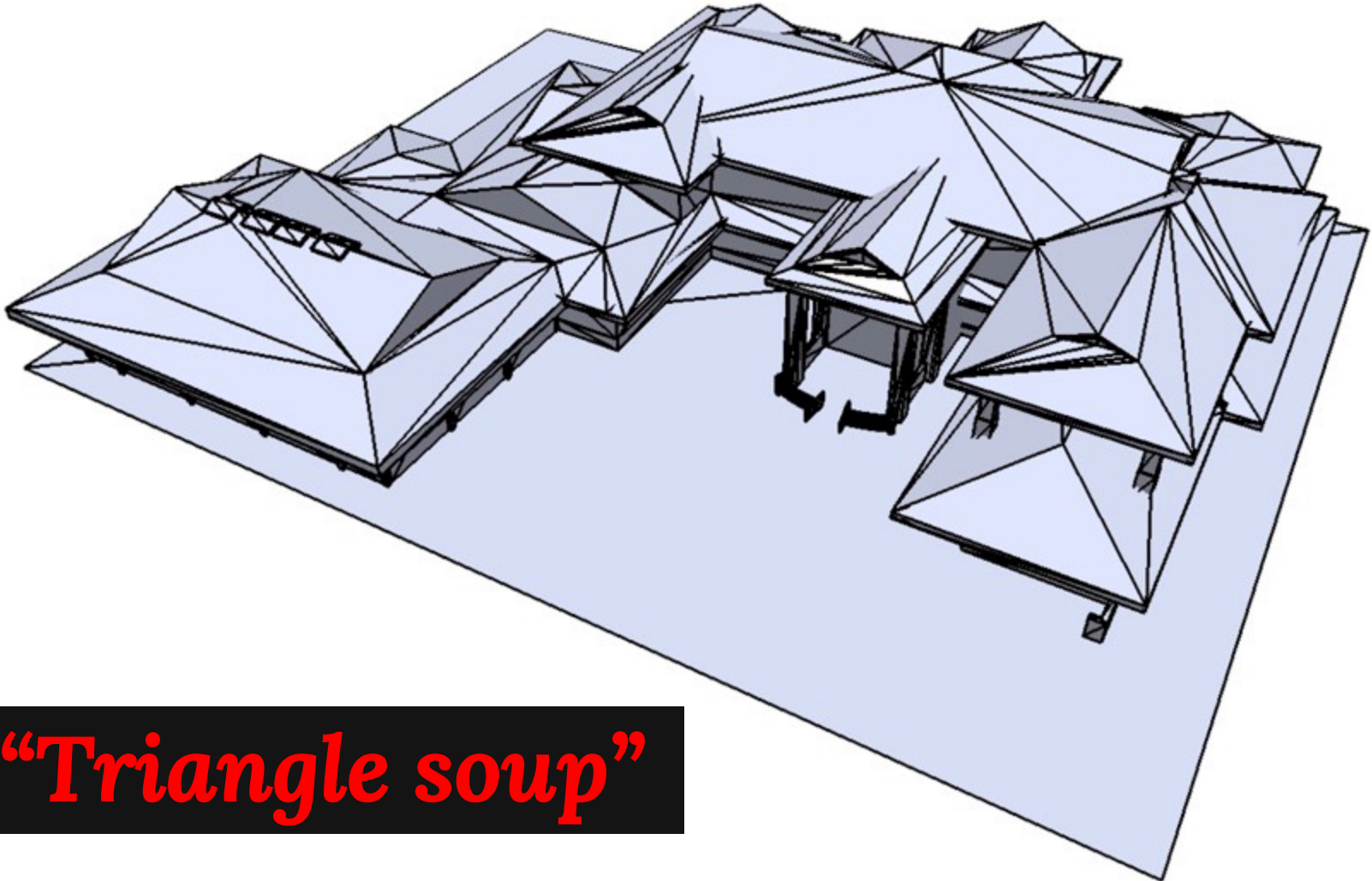
MANIFOLD MESH

1. Each **edge** is incident to one or two faces
2. **Faces** incident to a vertex form a closed or open fan

Assume meshes are manifold
(for now)

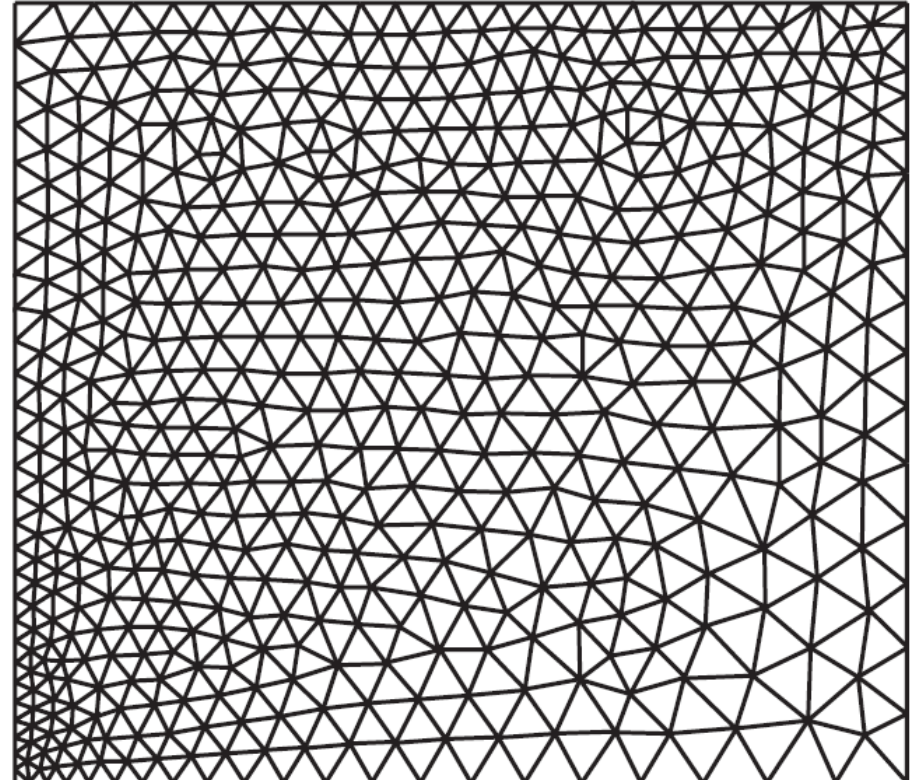
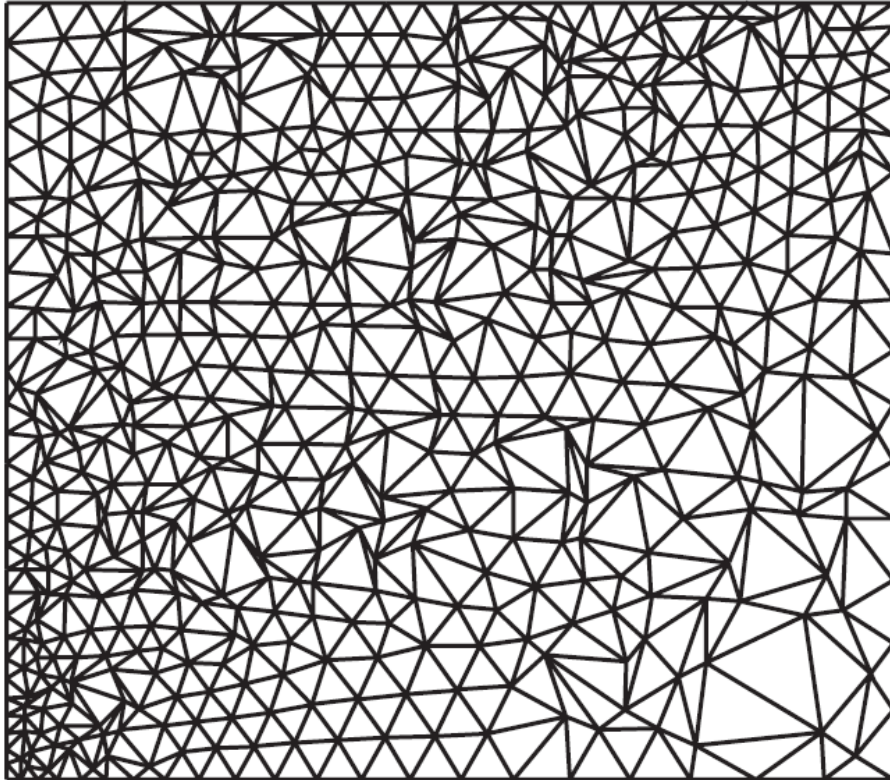


EASY-TO-VIOLATE ASSUMPTION



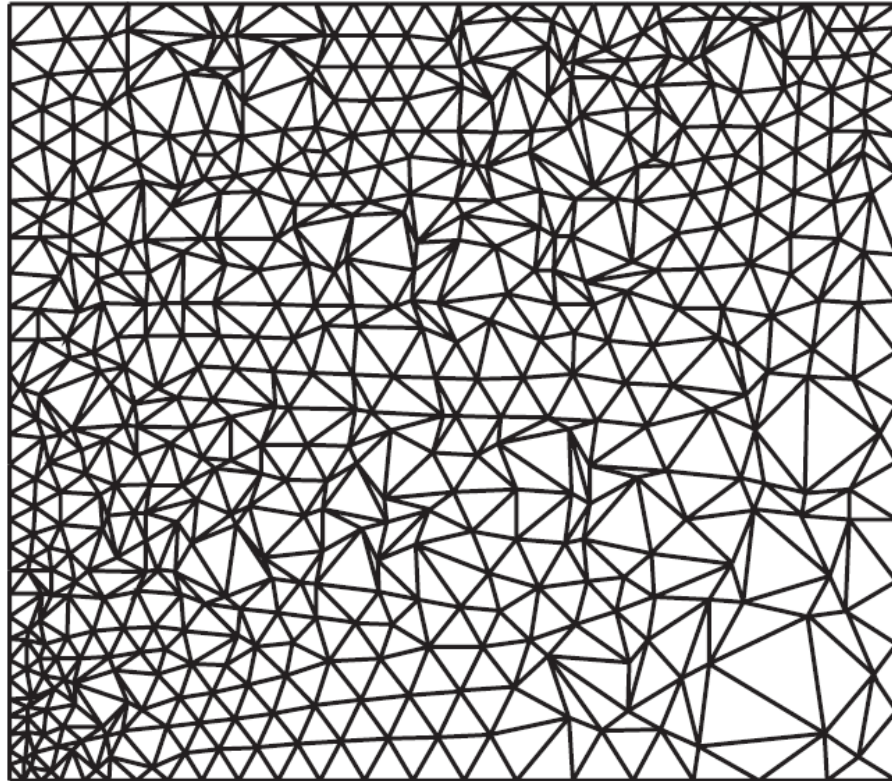
“Triangle soup”

INVALID MESHES VS. BAD MESHES



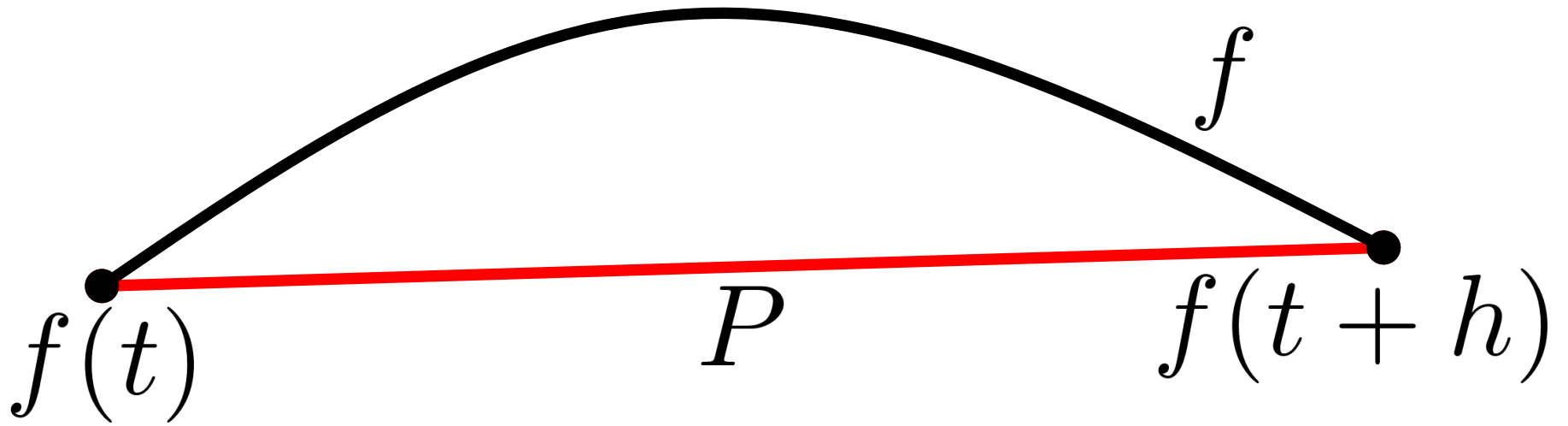
Nonuniform
areas and angles

WHY IS MESHING AN ISSUE?



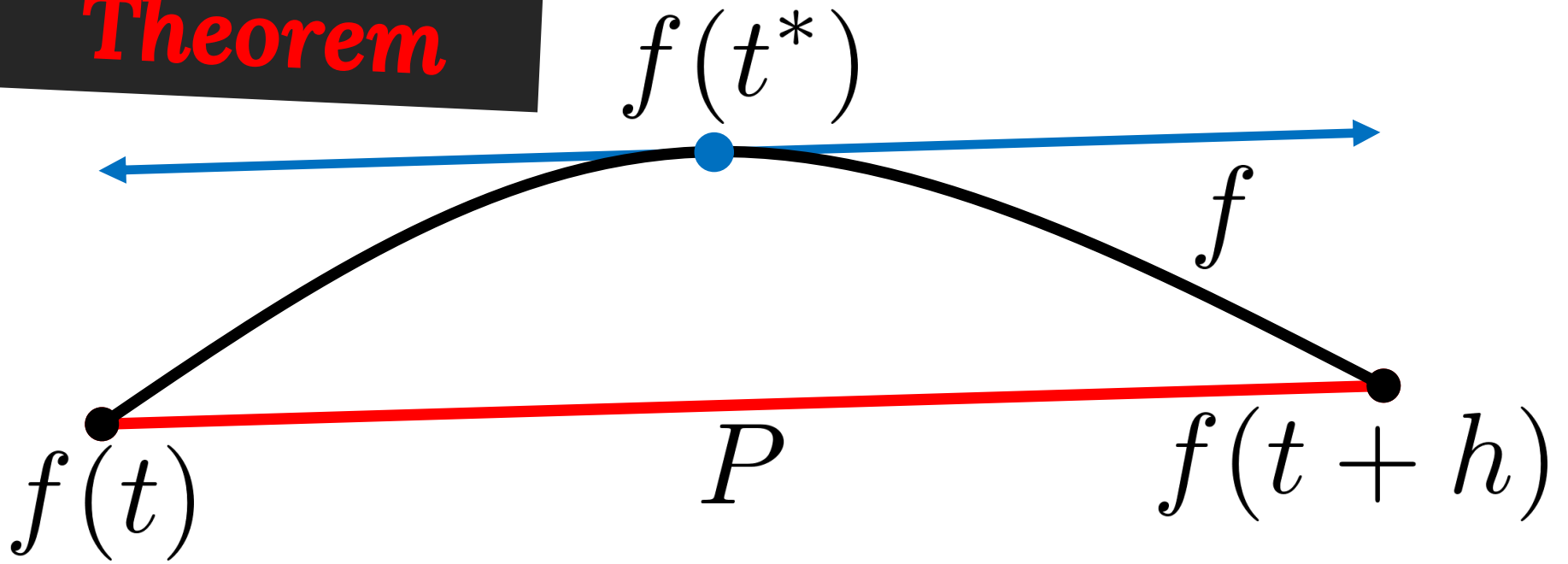
How to interpret
one value per vertex?

APPROXIMATION PROPERTIES

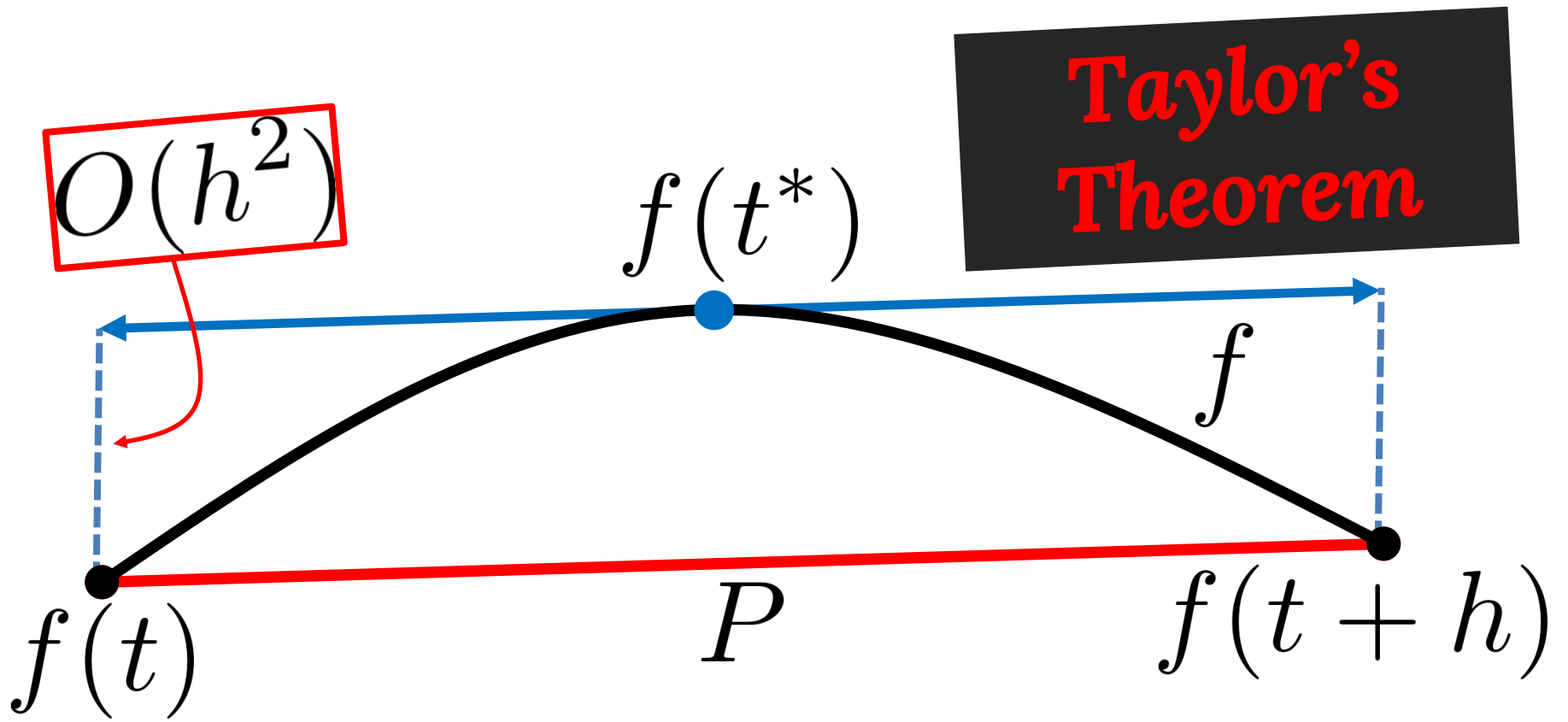


APPROXIMATION PROPERTIES

**Mean Value
Theorem**



APPROXIMATION PROPERTIES



CONCLUSION

*Piecewise linear faces are
reasonable **building blocks.***

ADDITIONAL ADVANTAGES

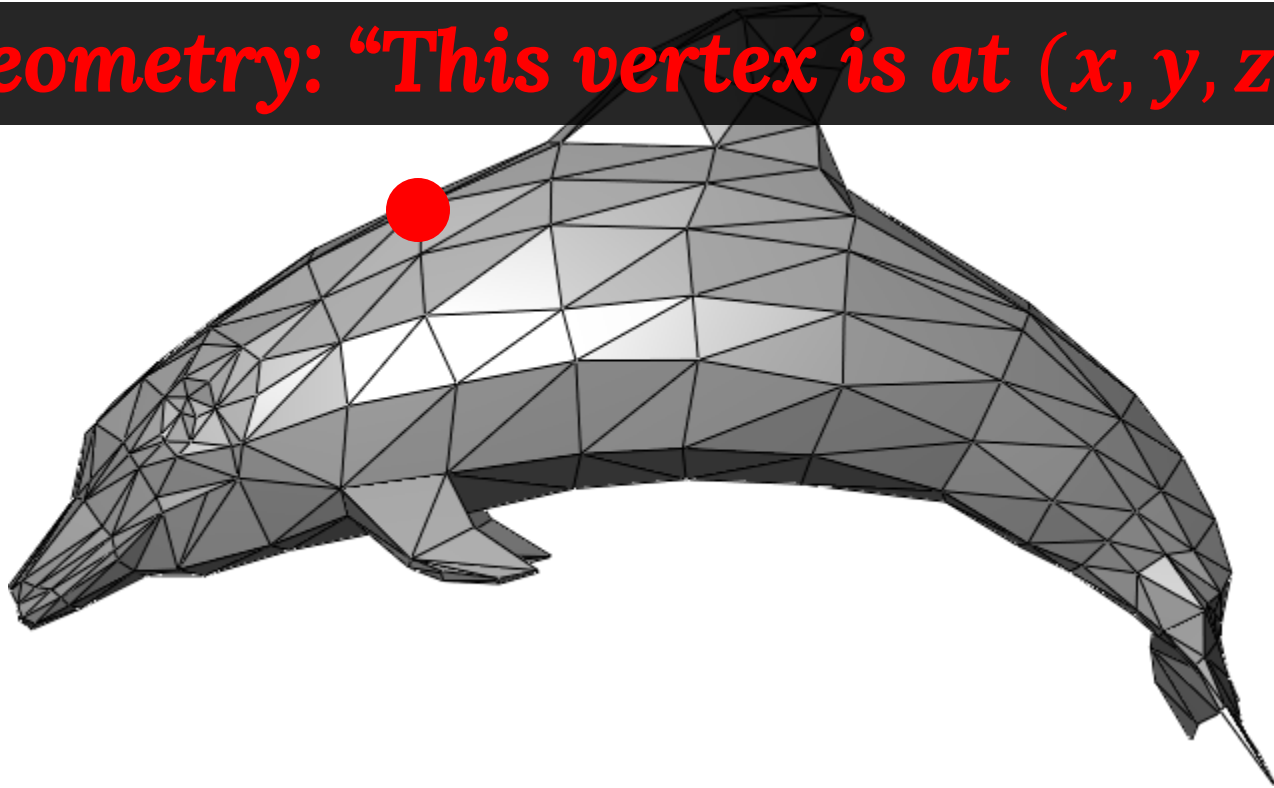
- Simple to render
- Arbitrary *topology* possible
- Basis for subdivision, refinement

Topology

The study of geometric properties that remain invariant under certain transformations

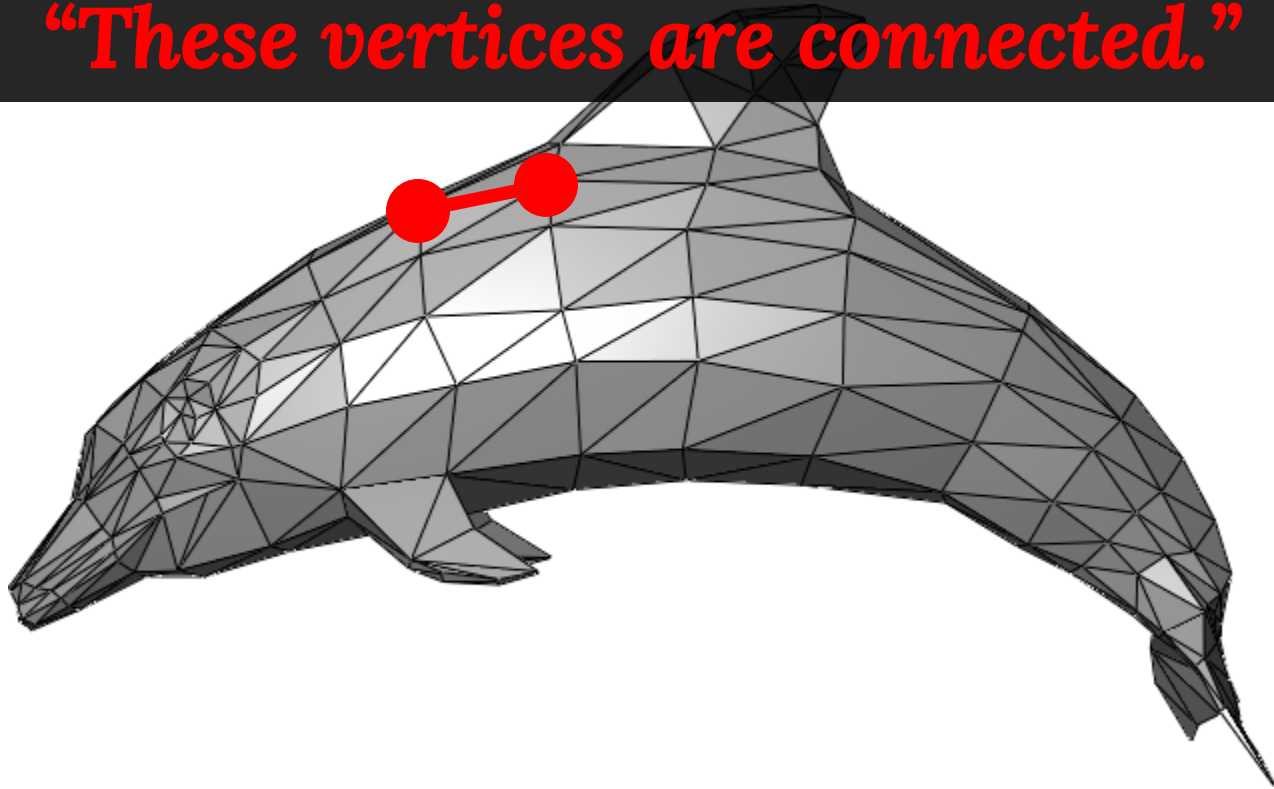
MESH TOPOLOGY VS. GEOMETRY

Geometry: "This vertex is at (x, y, z) ."



MESH TOPOLOGY VS. GEOMETRY

Topology:
“These vertices are connected.”



TRIANGLE MESH

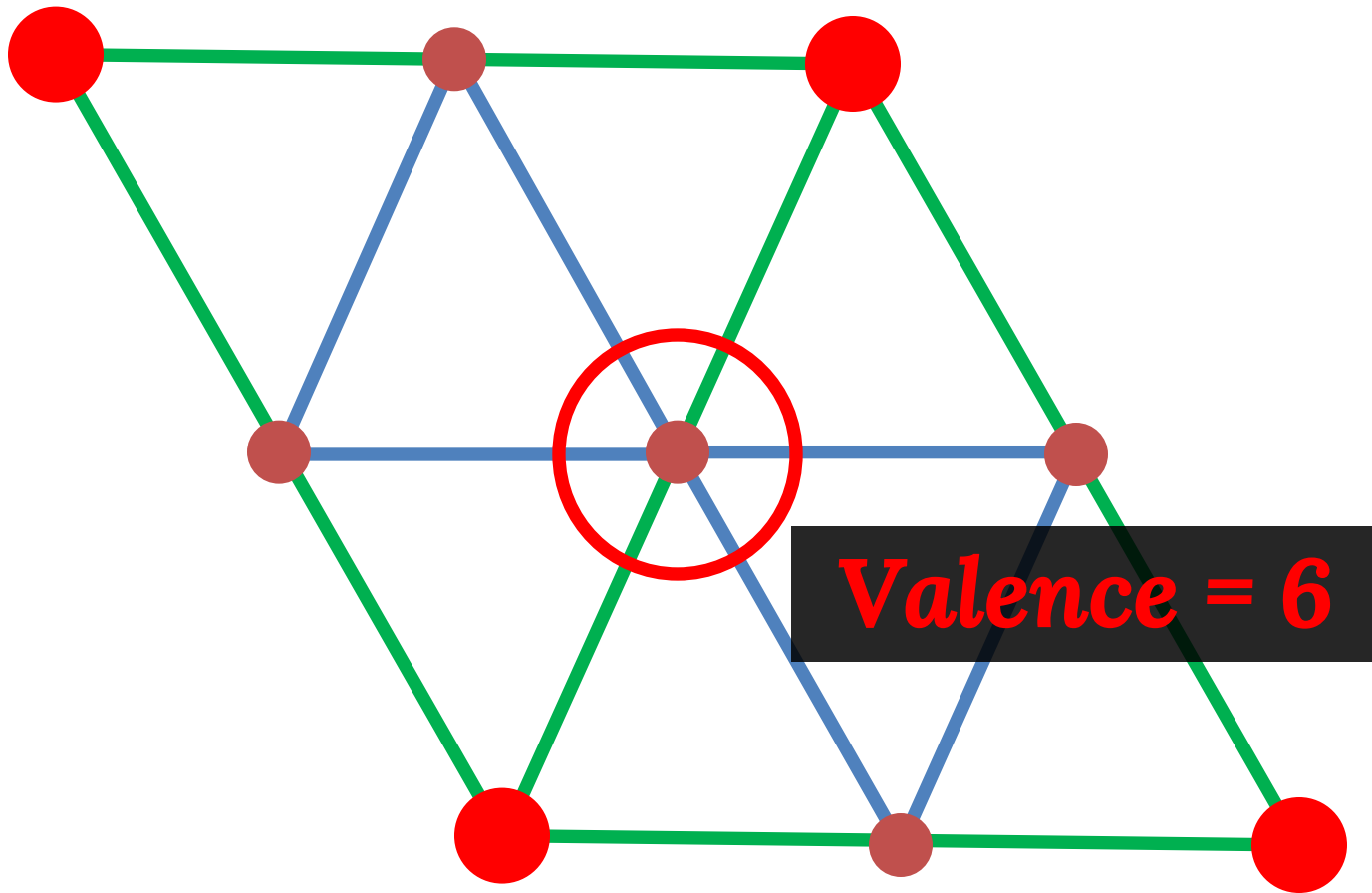
$$V = (v_1, v_2, \dots, v_n) \subset \mathbb{R}^n$$

$$E = (e_1, e_2, \dots, e_k) \subseteq V \times V$$

$$F = (f_1, f_2, \dots, f_m) \subseteq V \times V \times V$$

Plus manifold conditions

VALENCE



EULER CHARACTERISTIC

$$V - E + F := \chi$$

$$\chi = 2 - 2g$$

Defer proof



$$g = 0$$



$$g = 1$$

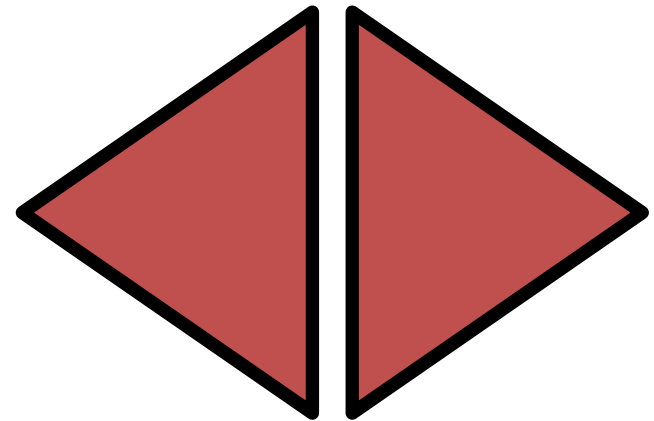


$$g = 2$$

CONSEQUENCES FOR TRIANGLE MESHES

$$V - E + F := \chi$$

“Each edge is adjacent to two faces. Each face has three edges.”



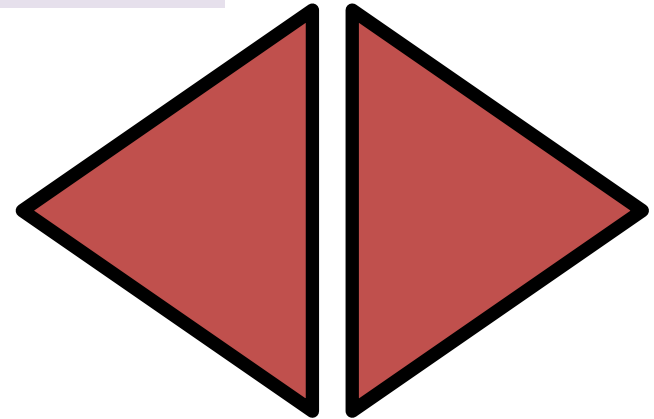
$$2E = 3F$$

Closed mesh: Easy estimates!

CONSEQUENCES FOR TRIANGLE MESHES

$$V - \frac{1}{2}F := \chi$$

“Each edge is adjacent to two faces. Each face has three edges.”



$$2E = 3F$$

Closed mesh: Easy estimates!

CONSEQUENCES FOR TRIANGLE MESHES

*Big
number*

$$V - \frac{1}{2}F := \chi$$

*Small
number*


$$F \approx 2V$$

Closed mesh: Easy estimates!

CONSEQUENCES FOR TRIANGLE MESHES

$$E \approx 3V$$

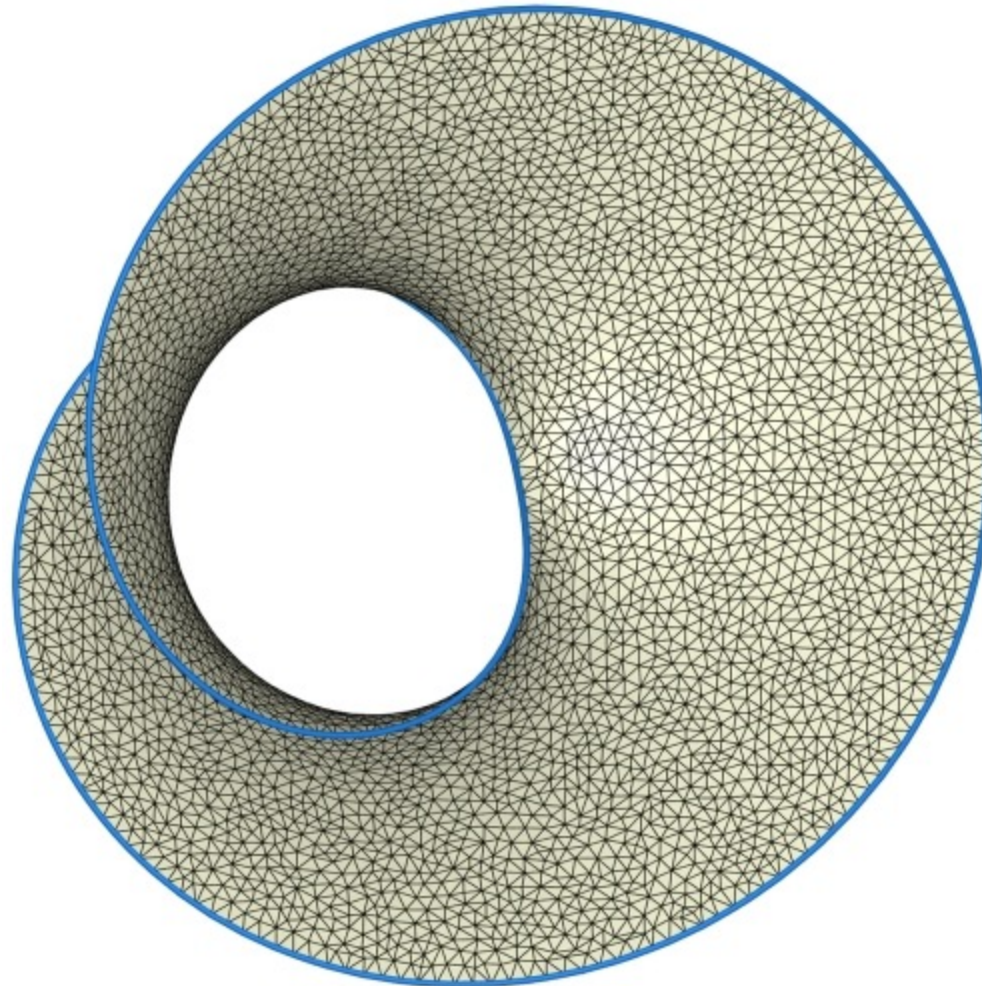
$$F \approx 2V$$

average valence ≈ 6

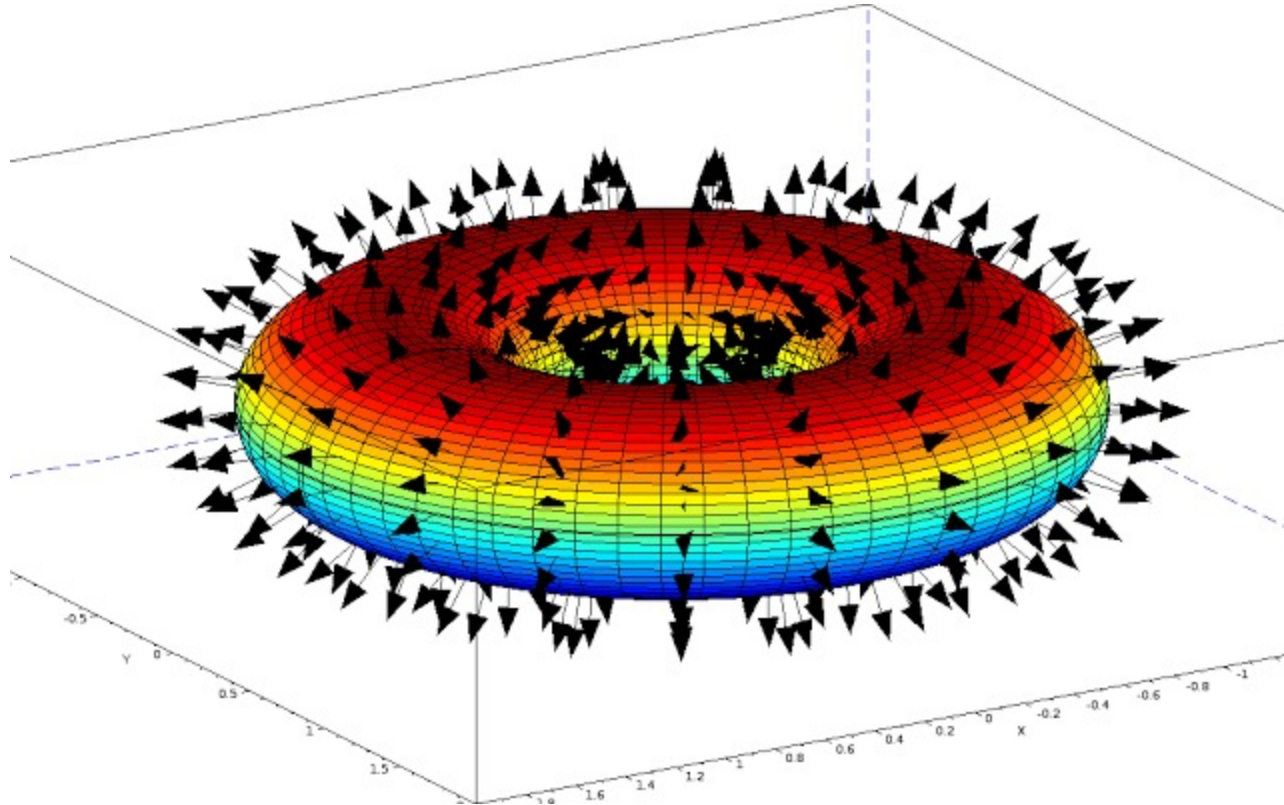
Why?!

General estimates

ORIENTABILITY



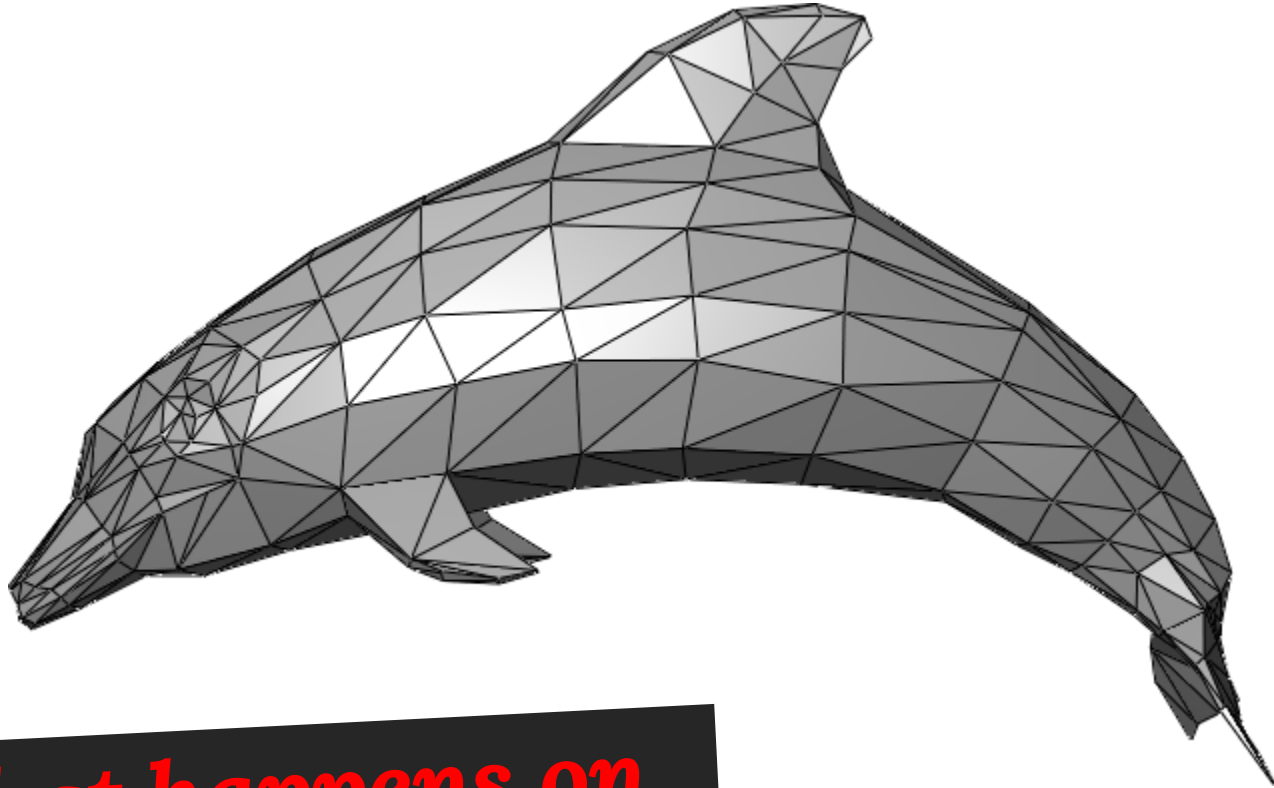
SMOOTH SURFACE DEFINITION



<https://lh3.googleusercontent.com/-njXPH7NSX5c/VV4PXu54n9I/AAAAAAAAAJM/m6TGg3ZVKGE/w640-h400-p->

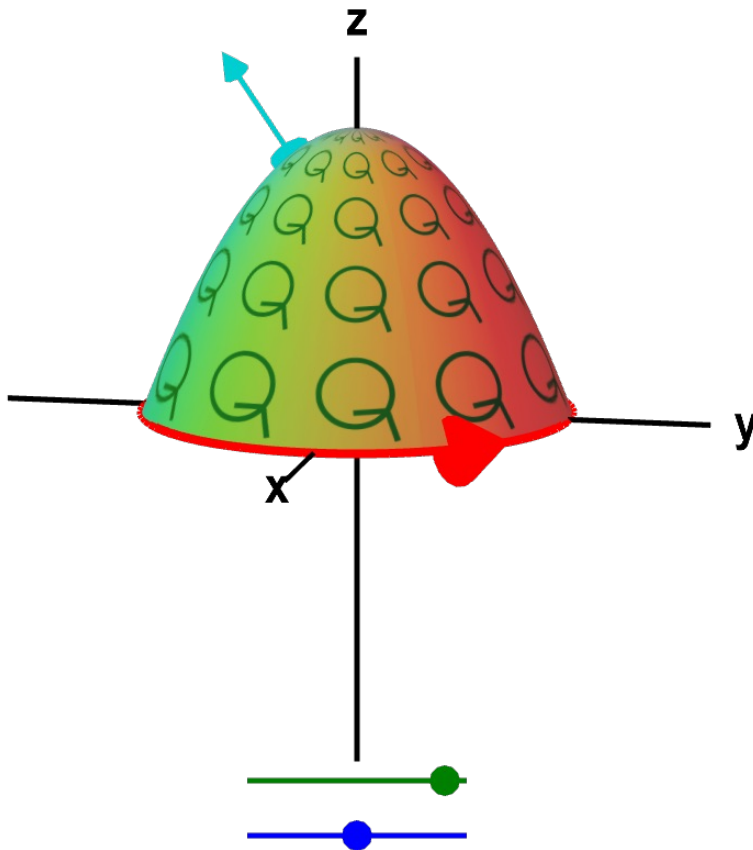
Continuous field of normal vectors

ISSUE ON TRIANGLE MESH

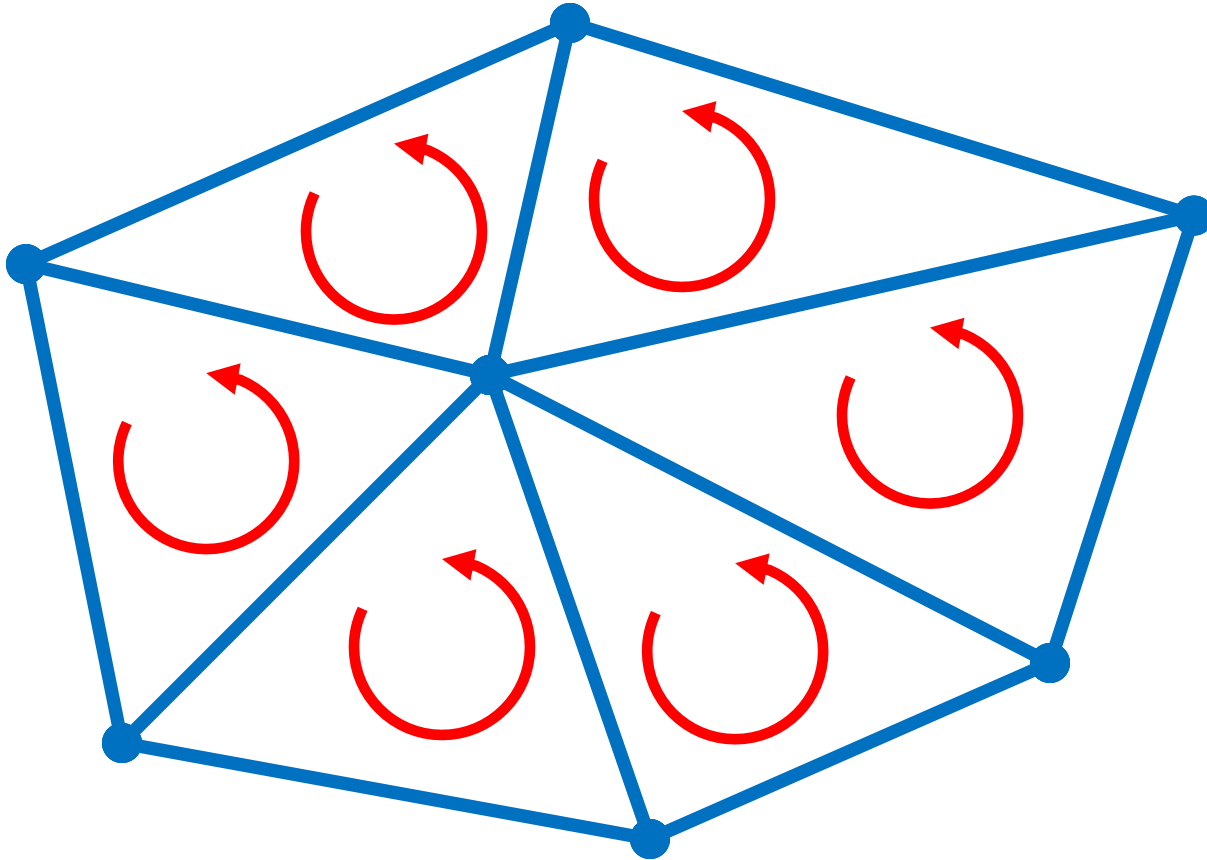


*What happens on
edges/vertices?*

RIGHT-HAND RULE



DISCRETE ORIENTABILITY



Normal field isn't continuous

DATA STRUCTURES FOR SURFACES

*Must represent geometry
and topology.*

SIMPLEST FORMAT

x1 y1 z1 / x2 y2 z2 / x3 y3 z3
x1 y1 z1 / x2 y2 z2 / x3 y3 z3
x1 y1 z1 / x2 y2 z2 / x3 y3 z3
x1 y1 z1 / x2 y2 z2 / x3 y3 z3
x1 y1 z1 / x2 y2 z2 / x3 y3 z3

No topology!

Triangle soup

SIMPLEST FORMAT

```
x1 y1 z1 / x2 y2 z2 / x3 y3 z3  
x1 y1 z1 / x2 y2 z2 / x3 y3 z3  
x1 y1 z1 / x2 y2 z2 / x3 y3 z3  
x1 y1 z1 / x2 y2 z2 / x3 y3 z3  
x1 y1 z1 / x2 y2 z2 / x3 y3 z3
```

No topology!

```
glBegin(GL_TRIANGLES)
```

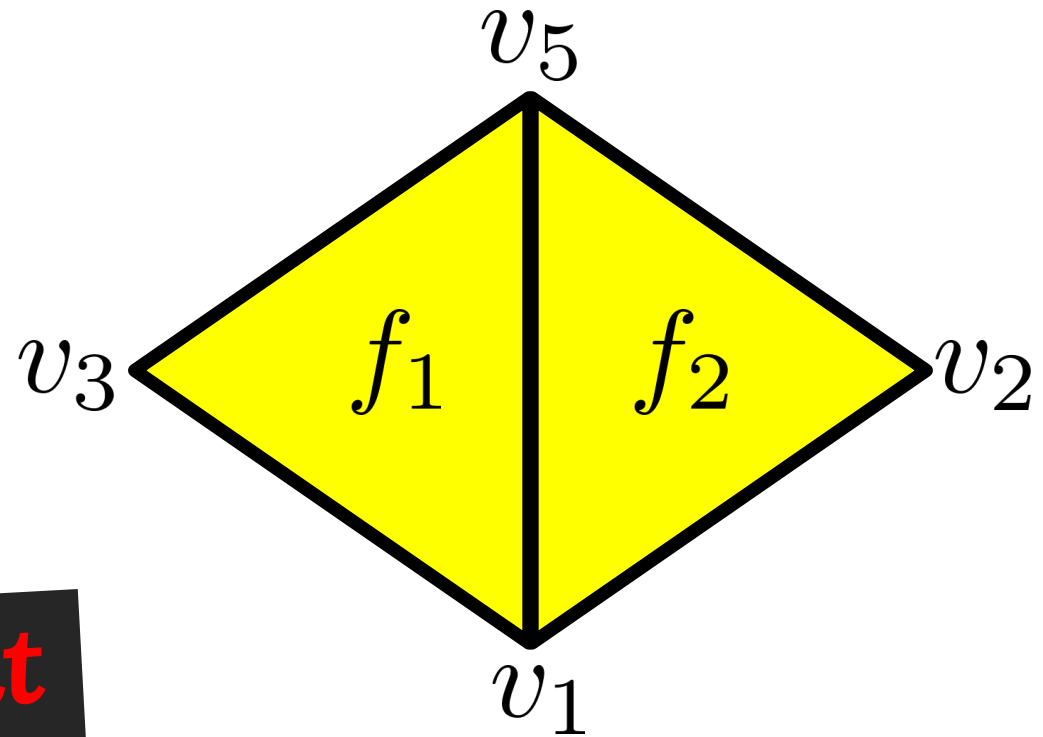
CS 468 2011 (M. Ben-Chen), other slides

Triangle soup

FACTOR OUT VERTICES

```
f 1 5 3
f 5 1 2
...
v 0.2 1.5 3.2
v 5.2 4.1 8.9
...
```

.obj format



Shared vertex structure

SIMPLE MESH SMOOTHING

```
for i=1 to n
  for each vertex v
    v = .5*v +
      .5*(average of neighbors);
```


TYPICAL QUERIES

- *Neighboring vertices to a vertex*
- *Neighboring faces to an edge*
- *Edges adjacent to a face*
- *Edges adjacent to a vertex*
- ...

Mostly localized

TYPICAL QUERIES

- **Neighboring** vertices to a vertex
- **Neighboring** faces to an edge
- Edges **adjacent** to a face
- Edges **adjacent** to a vertex
- ...

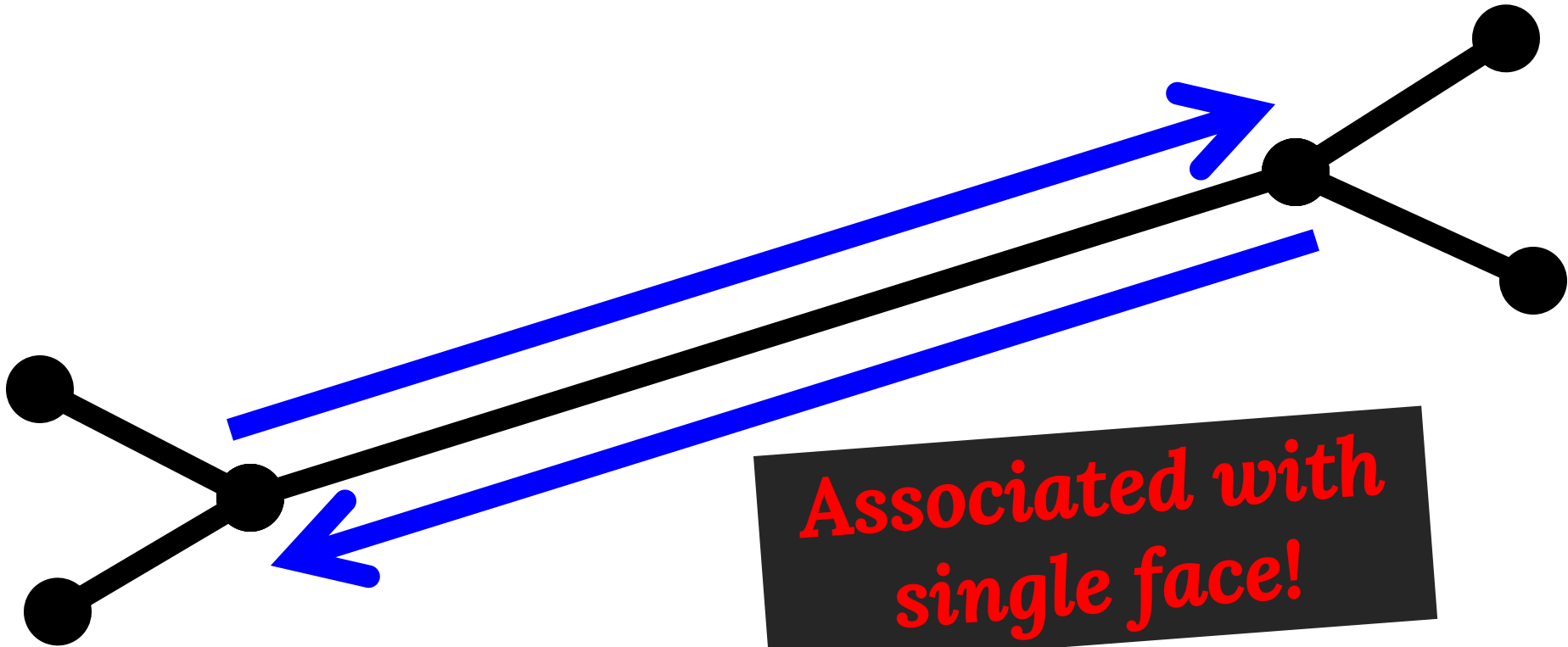
Mostly localized

PIECES OF HALFEDGE DATA STRUCTURE

- *Vertices*
- *Faces*
- *Half-edges*

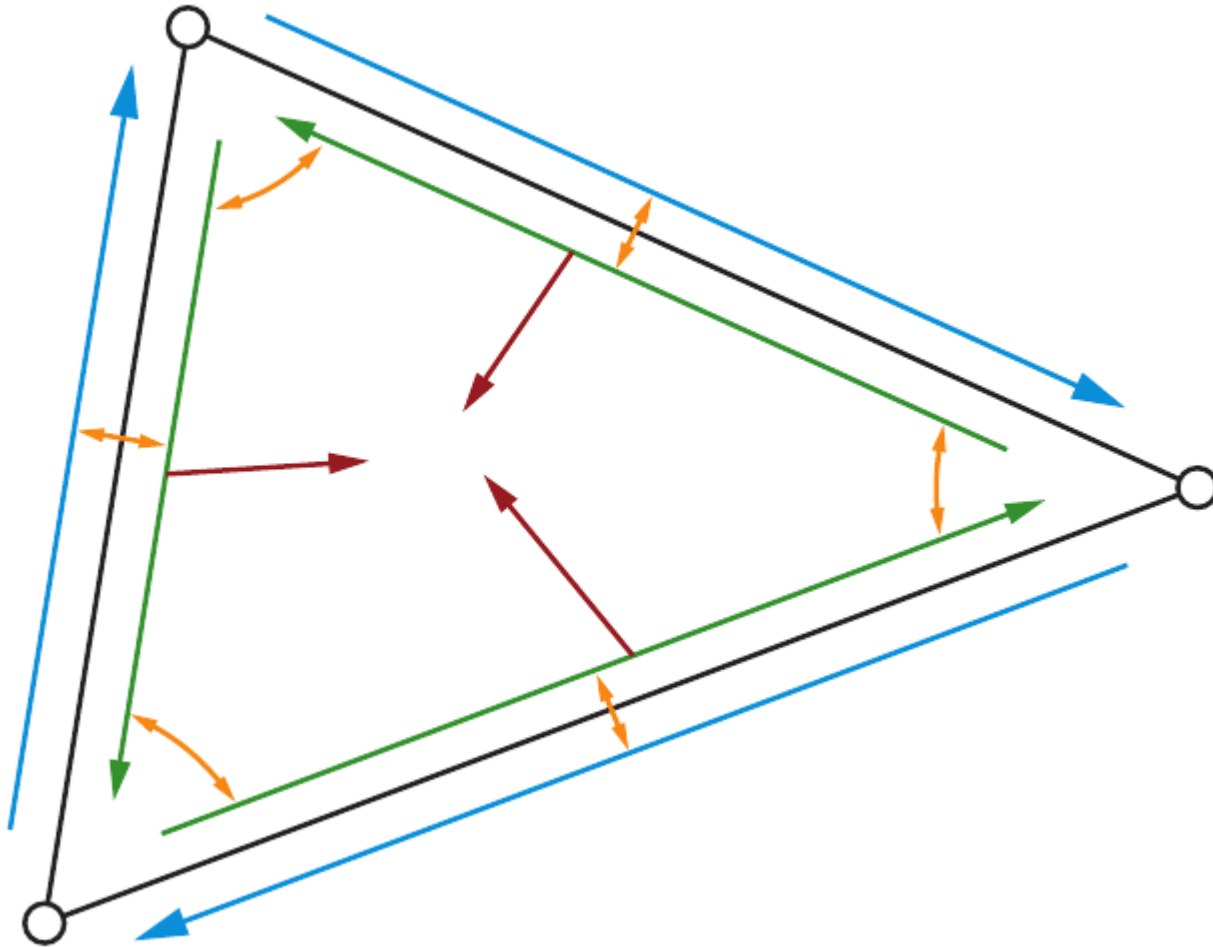
Structure tuned for meshes

HALFEDGE?



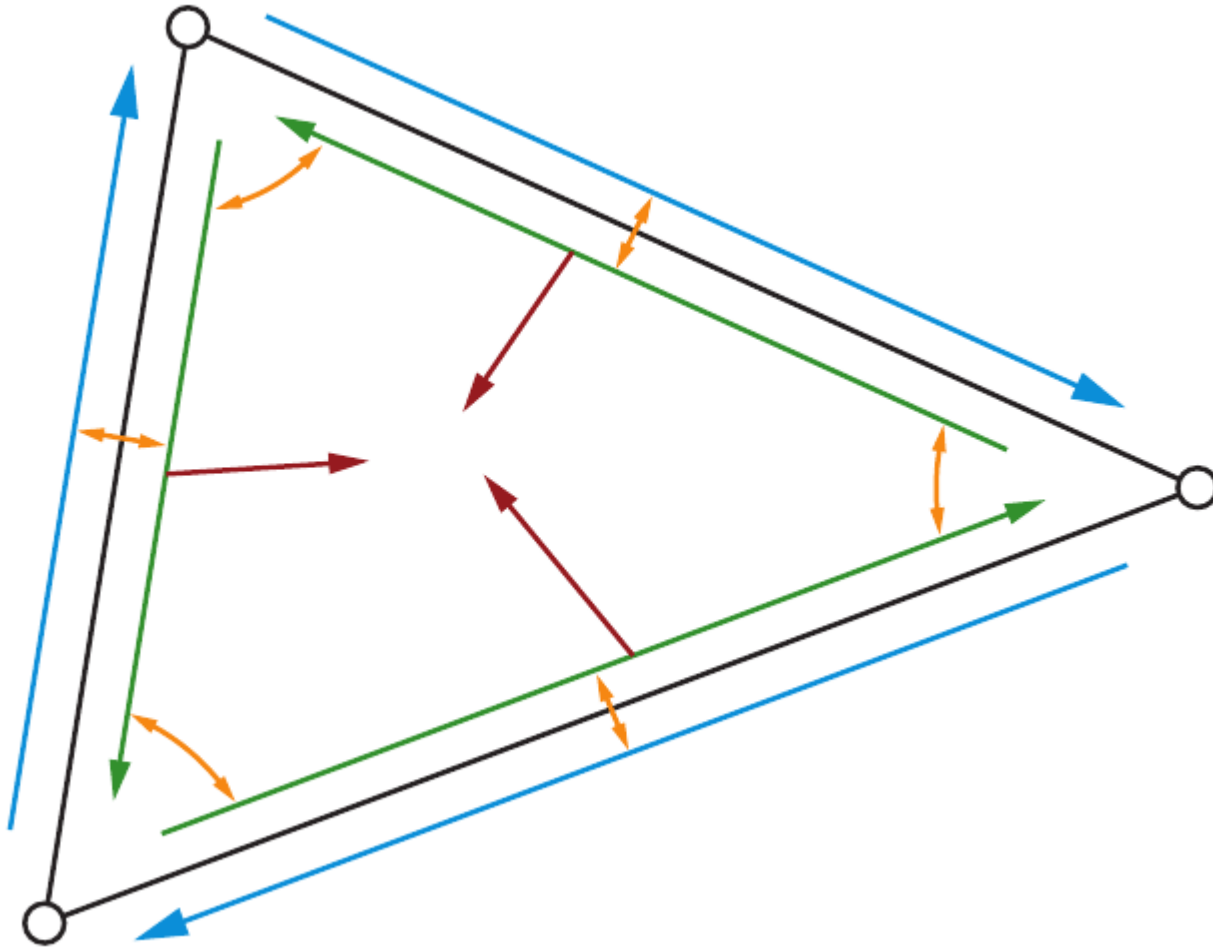
Oriented edge

HALFEDGE DATA TYPES



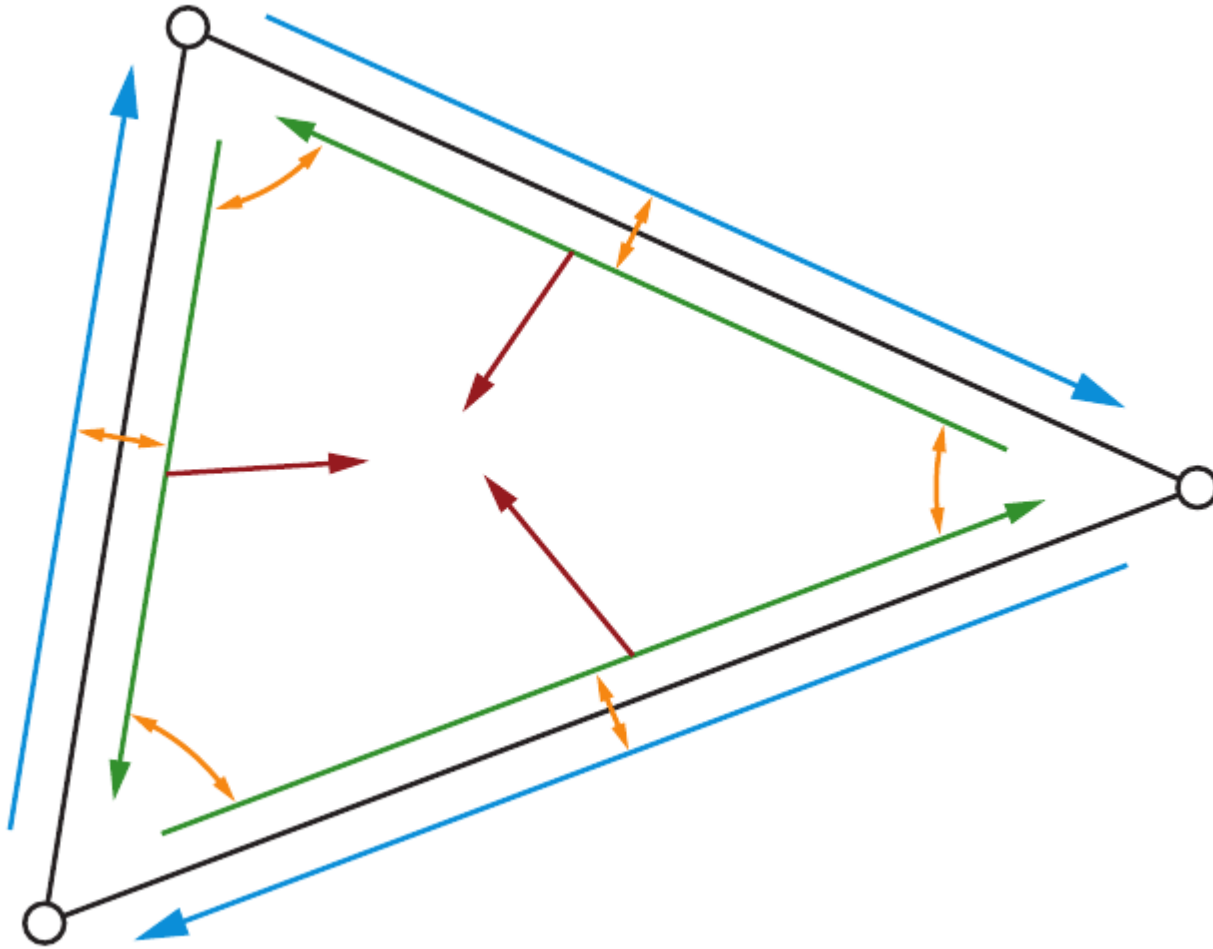
- Vertex stores:**
- **Arbitrary outgoing halfedge**

HALFEDGE DATA TYPES



- Face** stores:
- **Arbitrary adjacent halfedge**

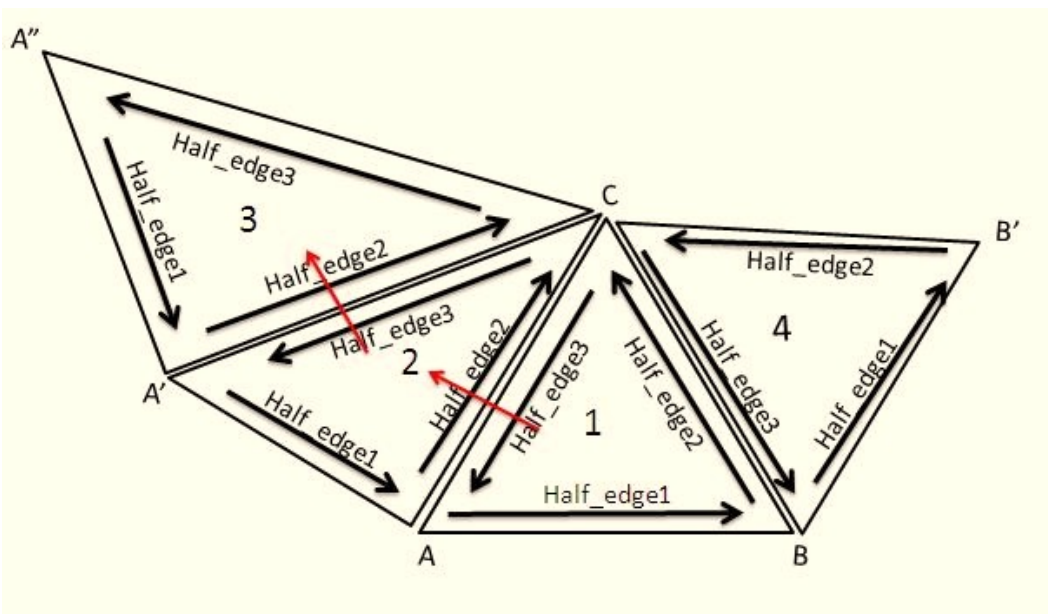
HALFEDGE DATA TYPES



Halfedge stores:

- **Flip**
- **Next**
- **Face**
- **Vertex**

ITERATING OVER VERTEX NEIGHBORS



```
Iterate(v) :  
startEdge = v.out;  
e = startEdge;  
do  
    process(e.flip.from)  
    e = e.flip.next  
while e != startEdge
```


ONLY SCRATCHING THE SURFACE

Eurographics Symposium on Geometry Processing (2005)
M. Desbrun, H. Pottmann (Editors)

Streaming Compression of Triangle Meshes

Martin Isenburg^{1†} Peter Lindstrom² Jack Snoeyink¹

¹ University of North Carolina at Chapel Hill ² Lawrence Livermore National Labs

EUROGRAPHICS 2011 / M. Chen and O. Deussen
(Guest Editors)

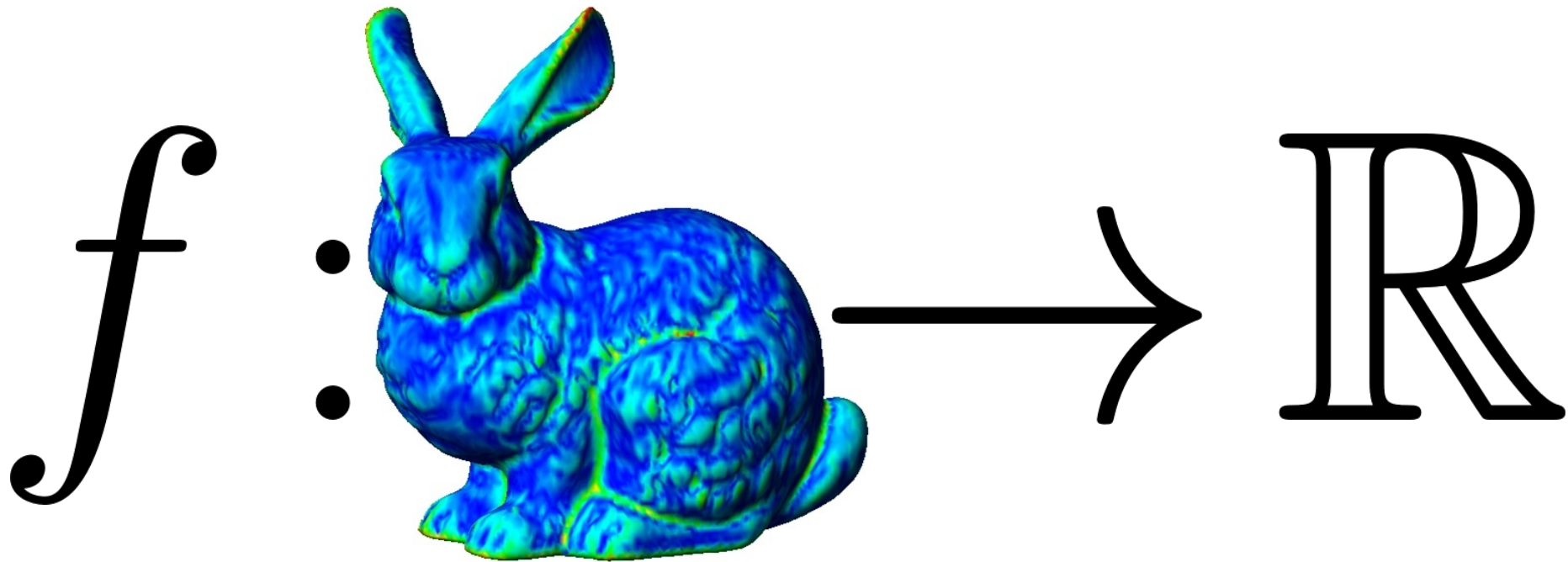
Volume 30 (2011), Number 2

SQuad: Compact Representation for Triangle Meshes

Topraj Gurung¹, Daniel Laney², Peter Lindstrom², Jarek Rossignac¹

¹Georgia Institute of Technology
²Lawrence Livermore National Laboratory

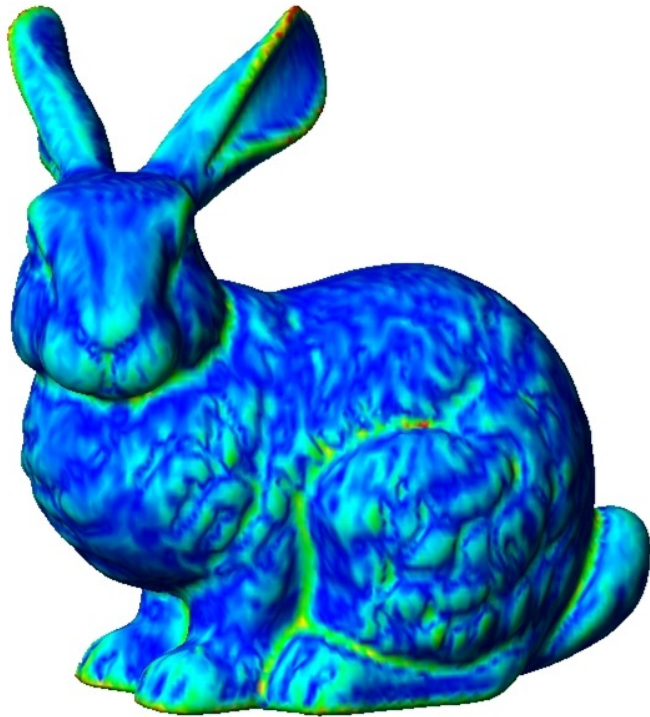
SCALAR FUNCTIONS



http://www.ieeta.pt/polymeco/Screenshots/PolyMeCo_OneView.jpg

Map points to real numbers

DISCRETE SCALAR FUNCTIONS



$$f \in \mathbb{R}^{|V|}$$

http://www.ieeta.pt/polymeco/Screenshots/PolyMeCo_OneView.jpg

Map vertices to real numbers

QUESTION

What is the integral of f ?

$$\int_M f \, dA$$

FINITE ELEMENTS APPROACH

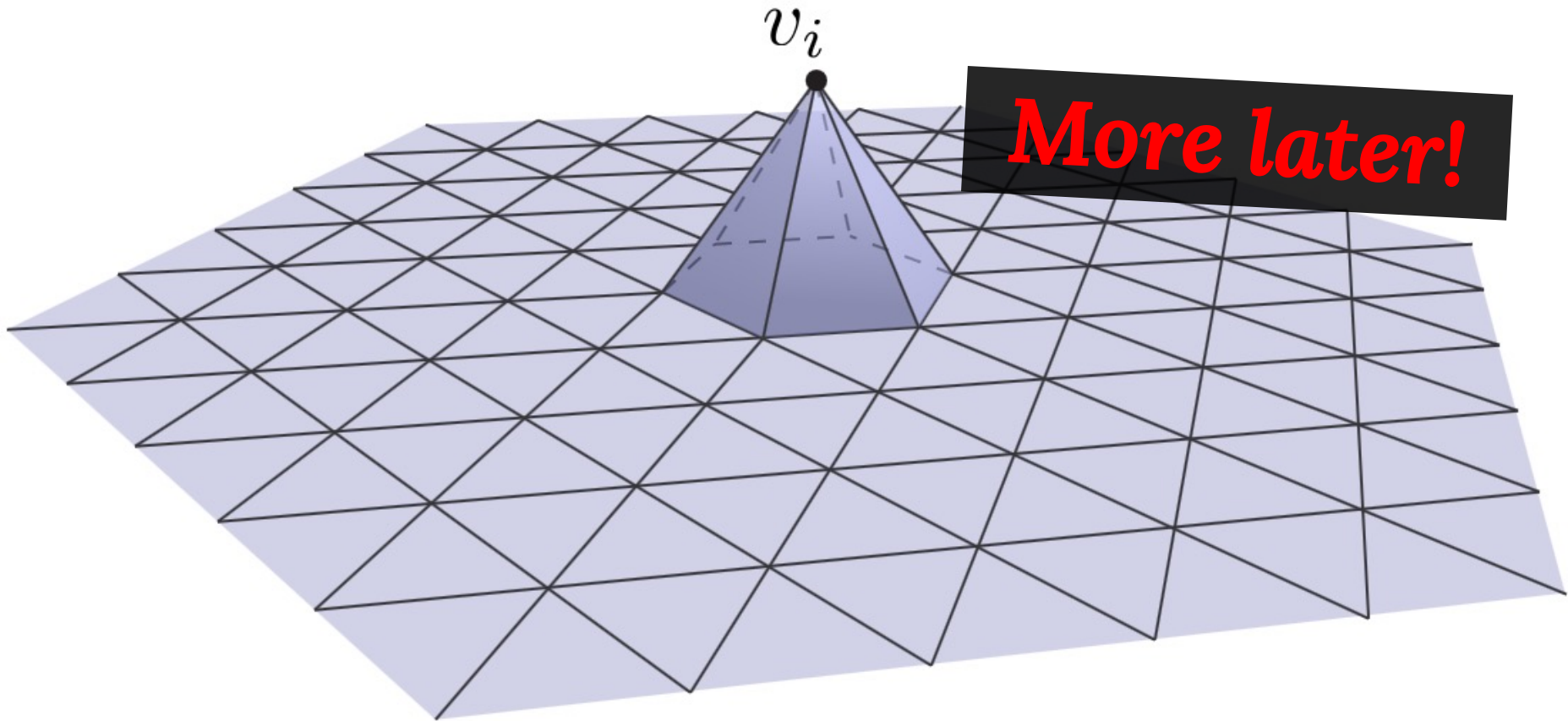
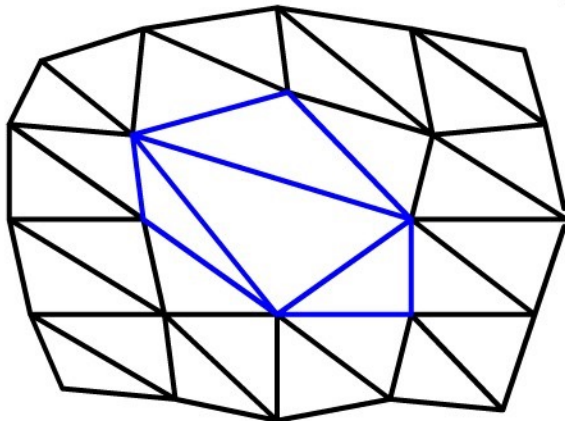
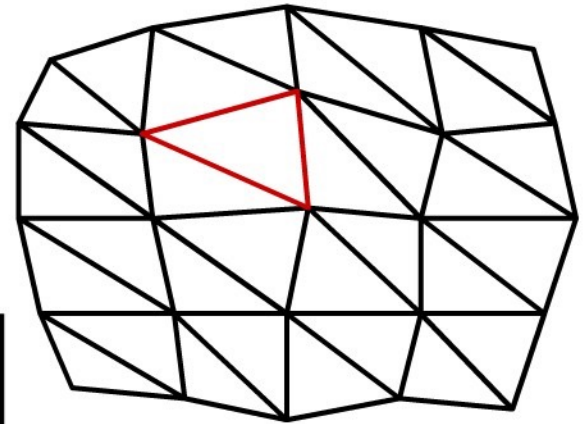
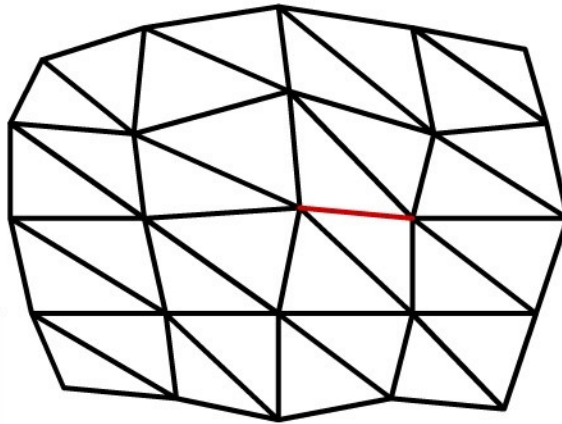
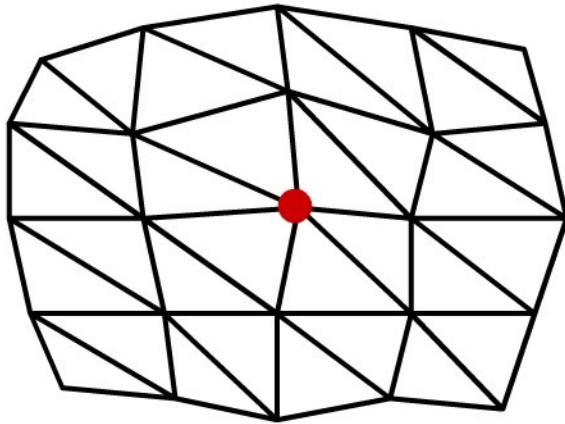


Image courtesy K. Crane

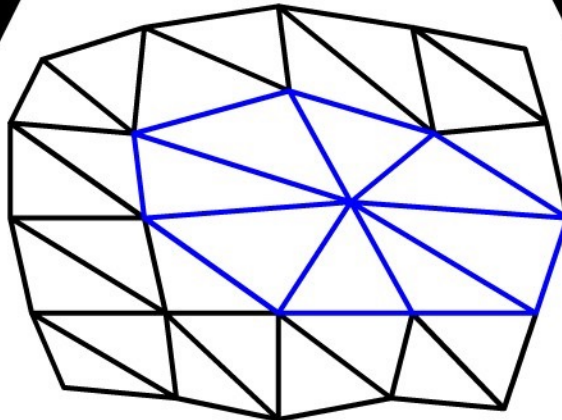
Use hat functions to interpolate

TOPOLOGICAL OPERATIONS

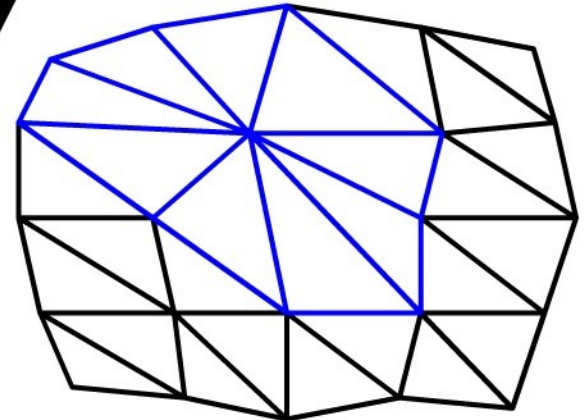
Original Mesh Segment



Vertex Removal



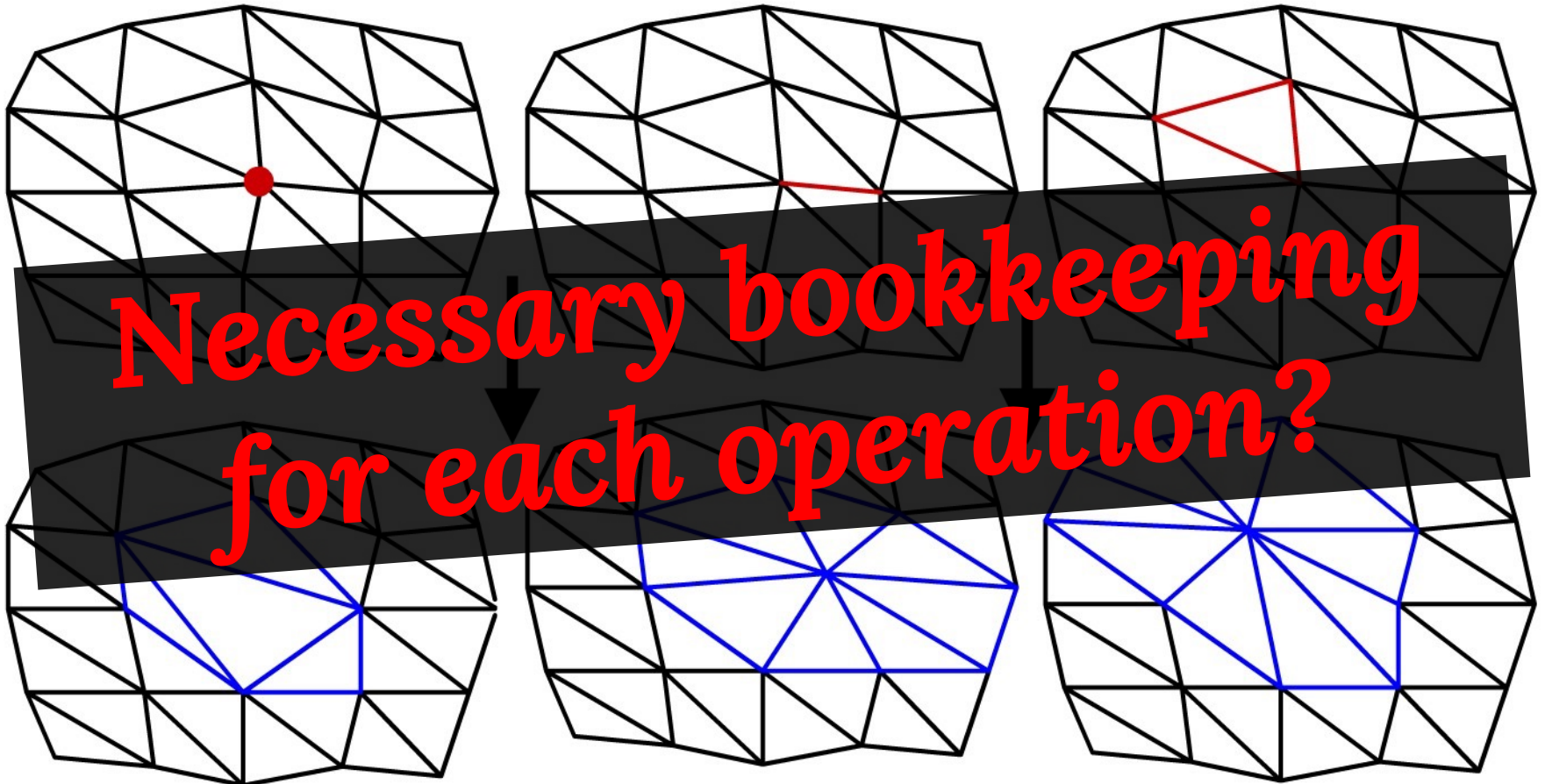
Edge Collapse



Face Collapse

TOPOLOGICAL OPERATIONS

Original Mesh Segment



Vertex Removal

Edge Collapse

Face Collapse

TAKE-AWAY

Simple Data
Structures

Complex Data
structures



- Less memory
- Less efficient operations

- More memory
- More efficient operations

NOT THE ONLY MODEL

AN IMPLICIT SURFACE TENSION MODEL

J. I. Hochstein* and T. L. Williams**

The University of Memphis
Memphis, Tennessee

ABSTRACT

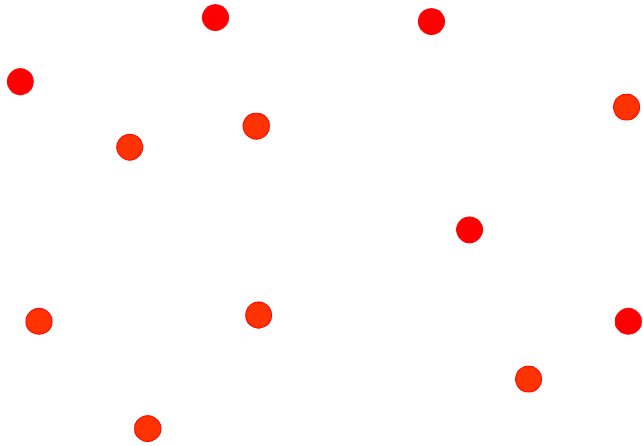
A new implicit model for surface tension at a two-fluid interface is proposed for use in computational models of flows with free surfaces and its performance is compared to an existing explicit model. The new model is based on an evolution equation for surface curvature that includes the influence of advection as well as surface tension. A detailed development of the new model is presented as are the details of the computational implementation. The performance of the new model is compared to an existing explicit model by using both models to predict the surface dynamics of several two-dimensional configurations. It is concluded that the new implicit surface tension model does perform better for configurations with a large surface tension coefficient. It is shown that, for several cases, the time step size is no longer limited by surface tension stability considerations (as it was using the explicit model), but rather by other limitations inherent in the existing volume advection algorithm.

INTRODUCTION

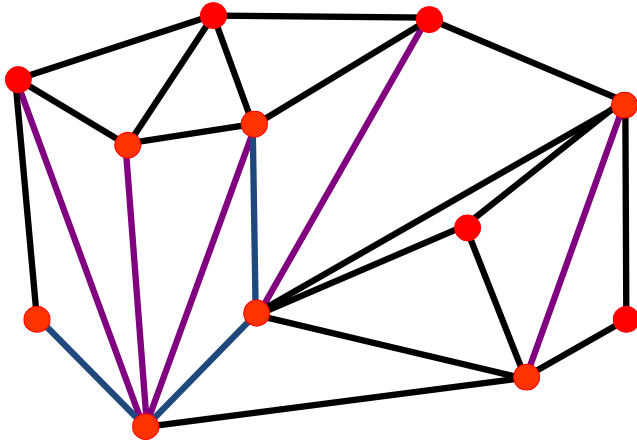
Incompressible flows with a free-surface exist in many industrial applications. Some examples include fuel atomization in internal combustion engines, droplet size control in ink-jet printers, formation of lead shot, control of liquid spacecraft propellant in low gravity, and the spinning of synthetic fibers. The technology for some of these applications has been developed by heavily relying on experimental study of the specific process involved. For others, such as spacecraft propellant management, experimental studies are prohibitively expensive and the ability to computationally model these process is essential for their development.

The modeling of flows with a free surface presents challenges unlike other types of flow problems in that a boundary condition must be applied at the free surface which is often in a transient state and irregularly shaped. This problem is exacerbated when the force due to surface tension

HOW TO GENERATE MESHES?

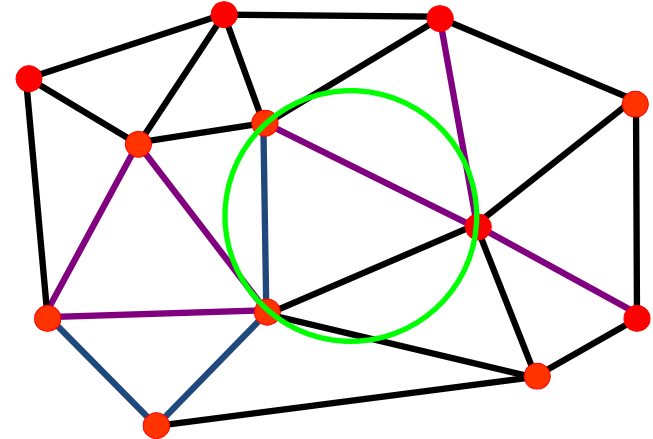
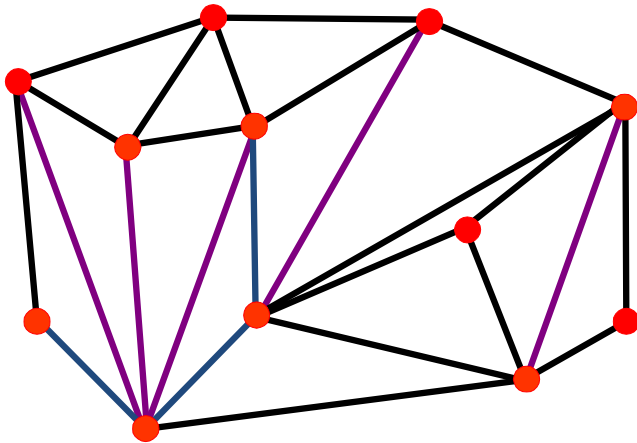


TRIANGULATION



Triangulation: triangle mesh on a set of points

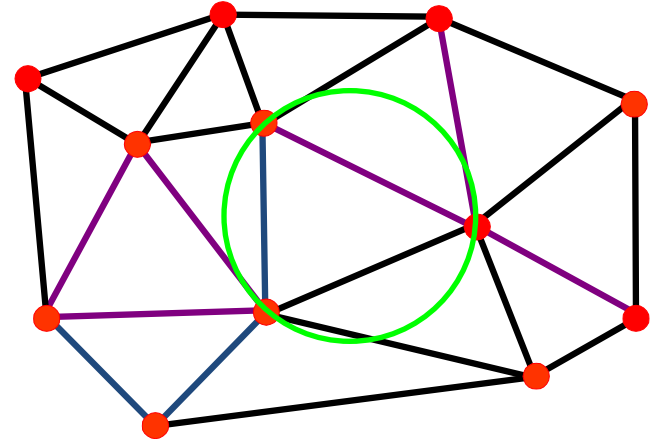
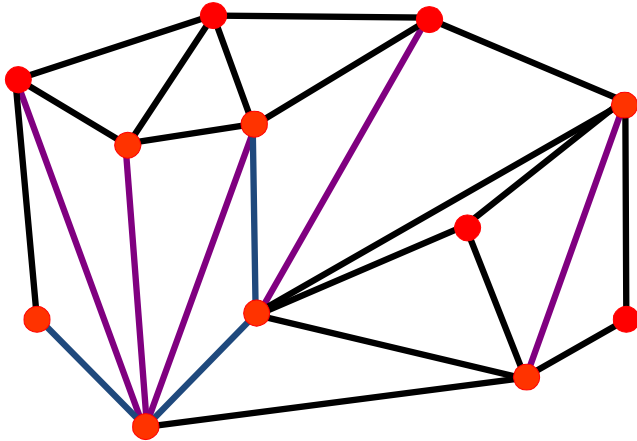
TRIANGULATION



Triangulation: triangle mesh on a set of points

Delaunay triangulation (DT):
a triangulation s.t. the circumcircle of any
triangle does not contain any other point.

FUN FACTS ABOUT DT

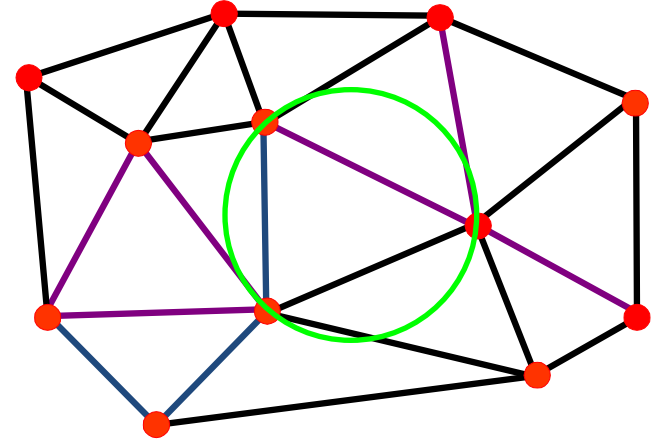
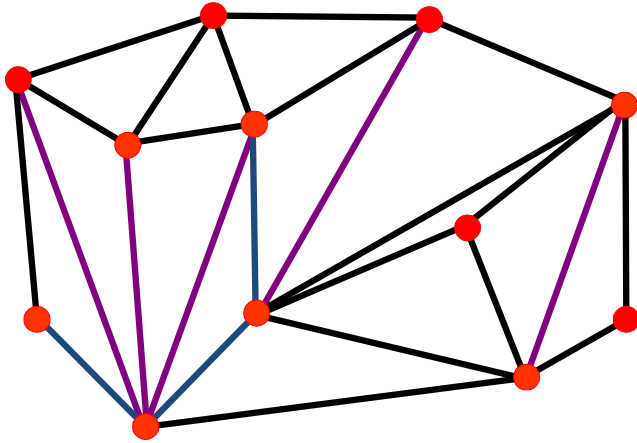


Theorem.

The exterior polygon of the DT is the convex hull.

DT is ***the best triangulation*** you can get in 2D.

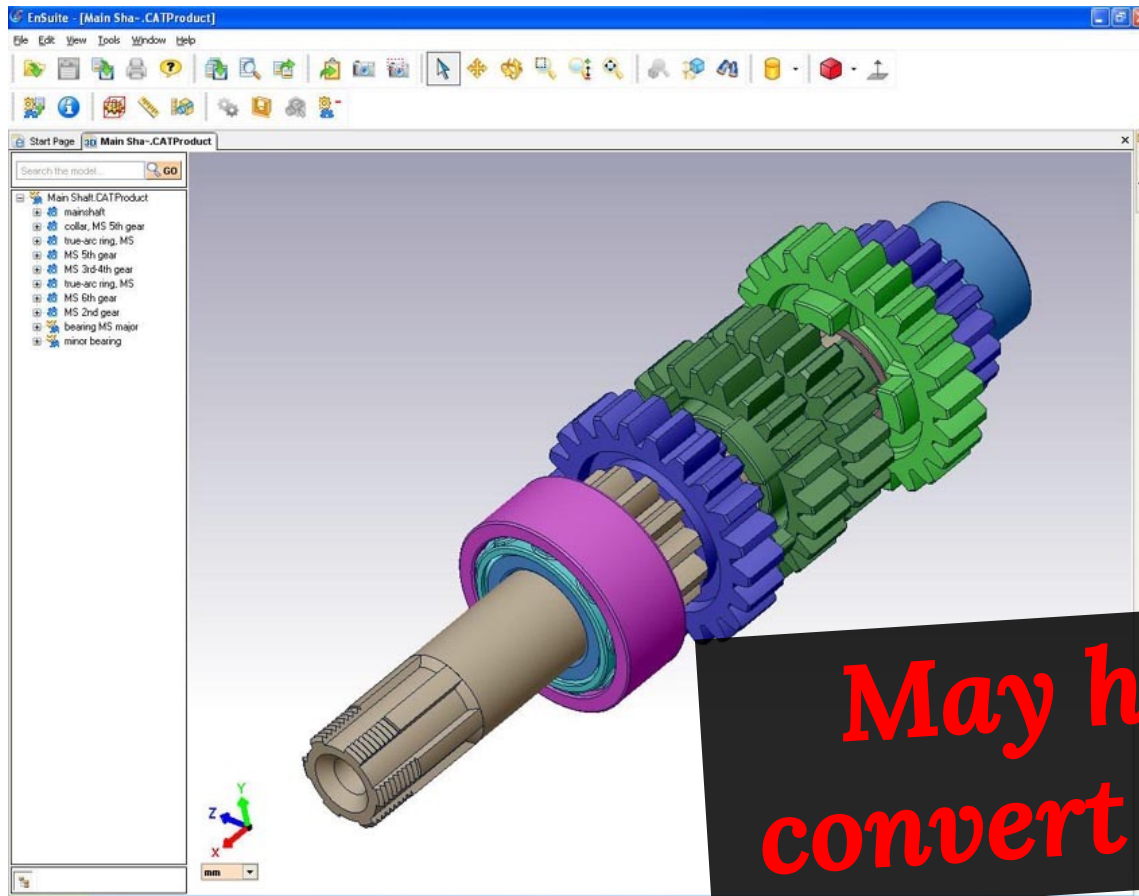
THE BEST *HOW*?



Take all angles, sort = Sequence of numbers
($9^\circ, 11^\circ, 12^\circ, 15^\circ, \dots$)

DT's *sequence* is ***the last lexicographically***

OBTAINING GEOMETRY

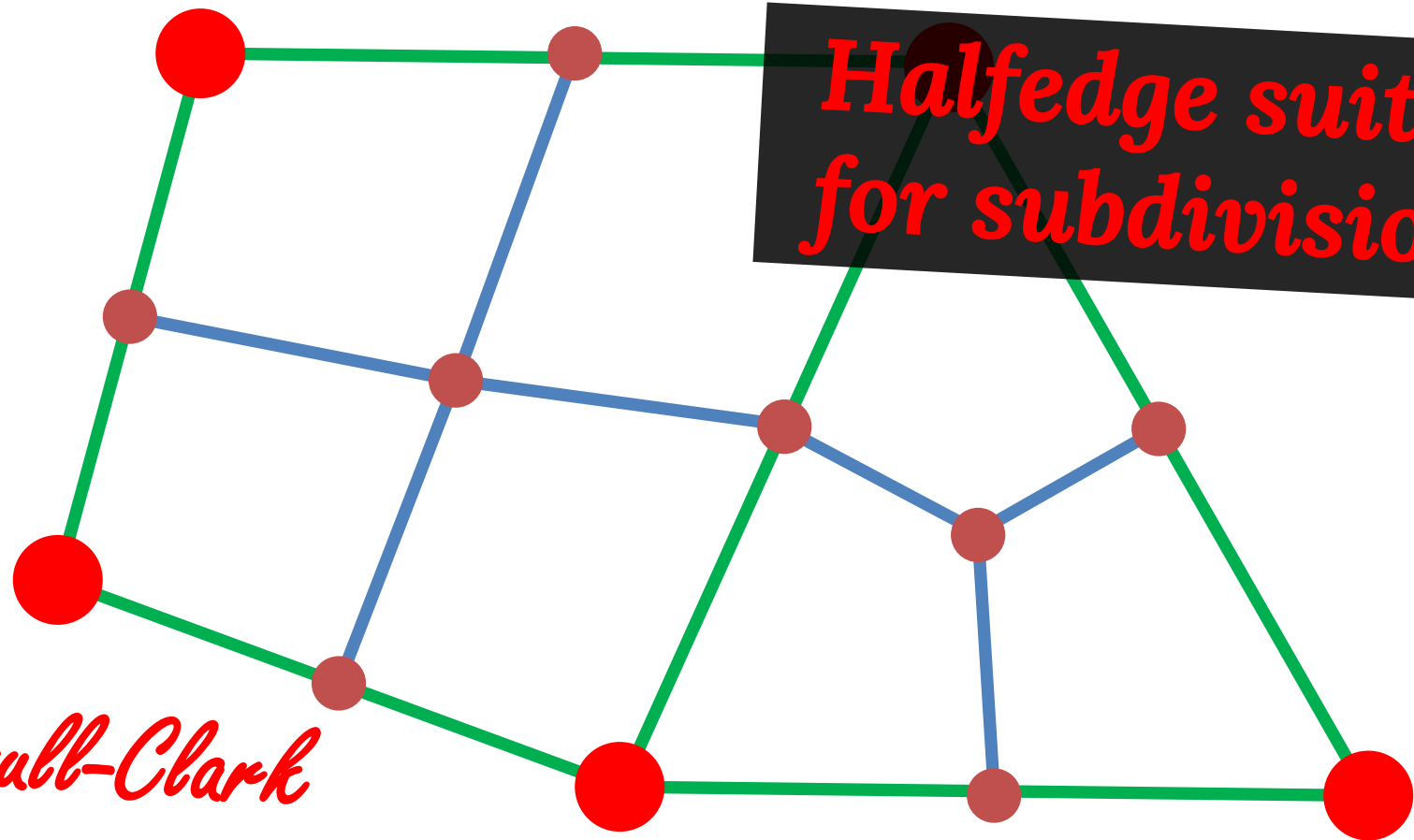


**May have to
convert to mesh**

<http://www.cad-sourcing.com/wp-content/uploads/2011/12/free-cad-software.jpg>

Cleanest: Design software

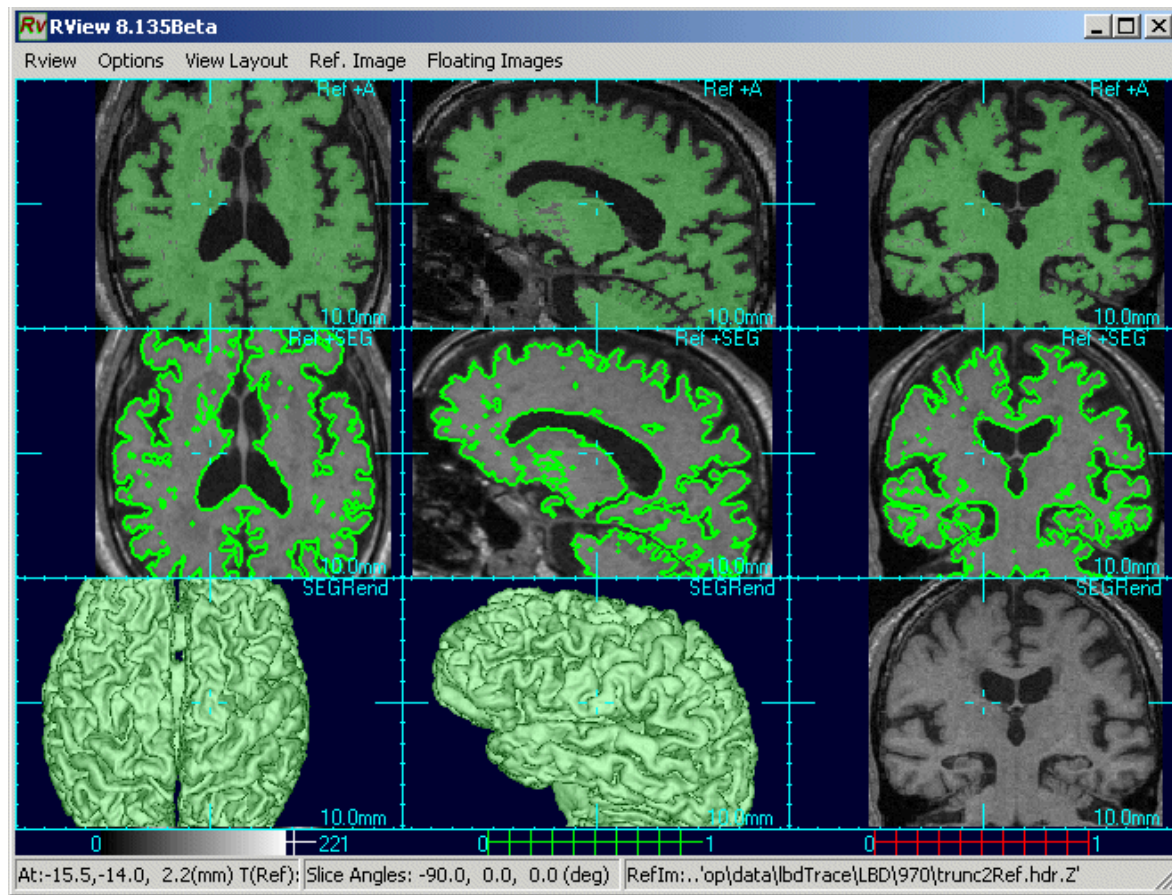
OBTAINING GEOMETRY



Catmull-Clark

Cleanest: Design software

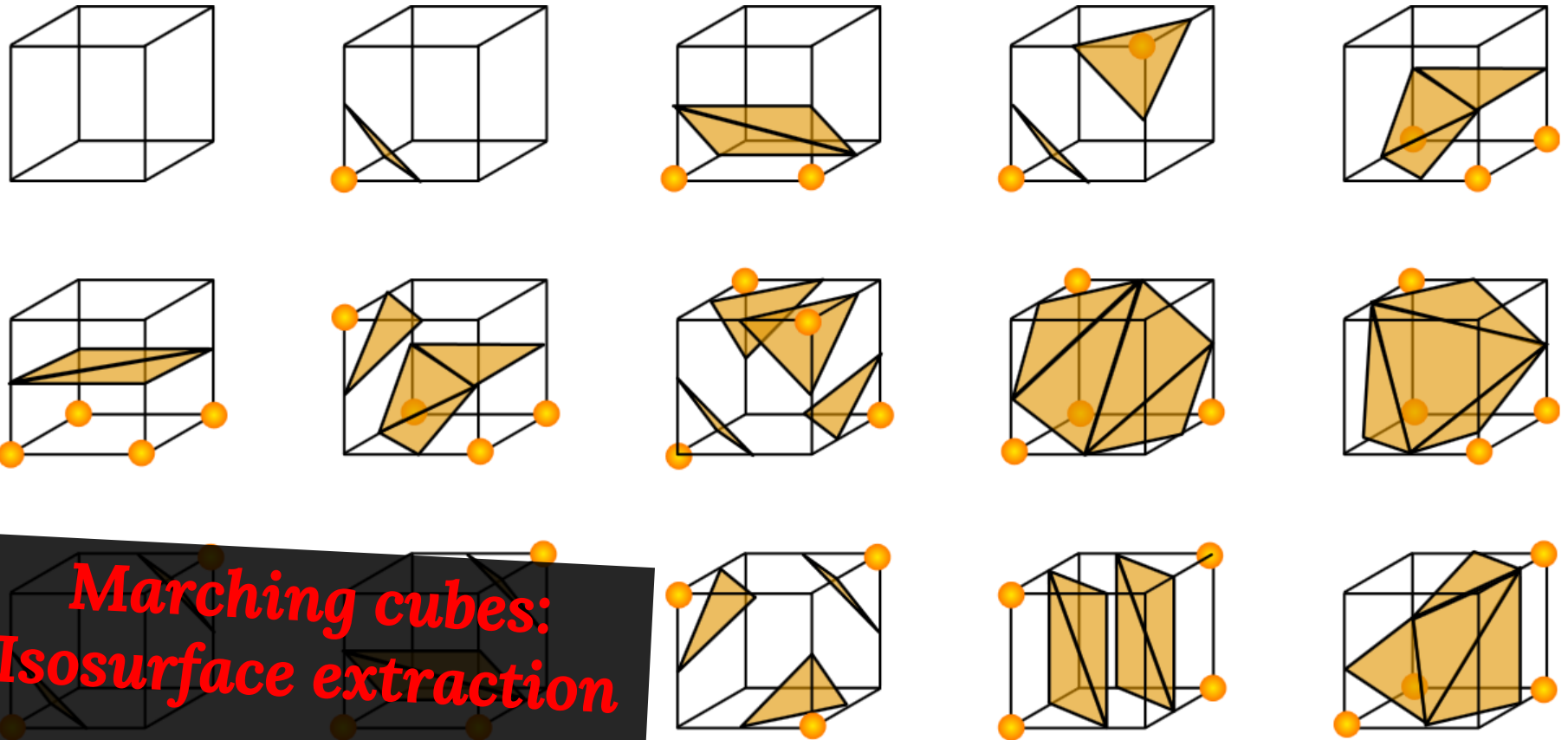
OBTAINING GEOMETRY



<http://www.colin-studholme.net/software/rview/rvmanual/morphool5.gif>

Volumetric extraction

OBTAINING GEOMETRY

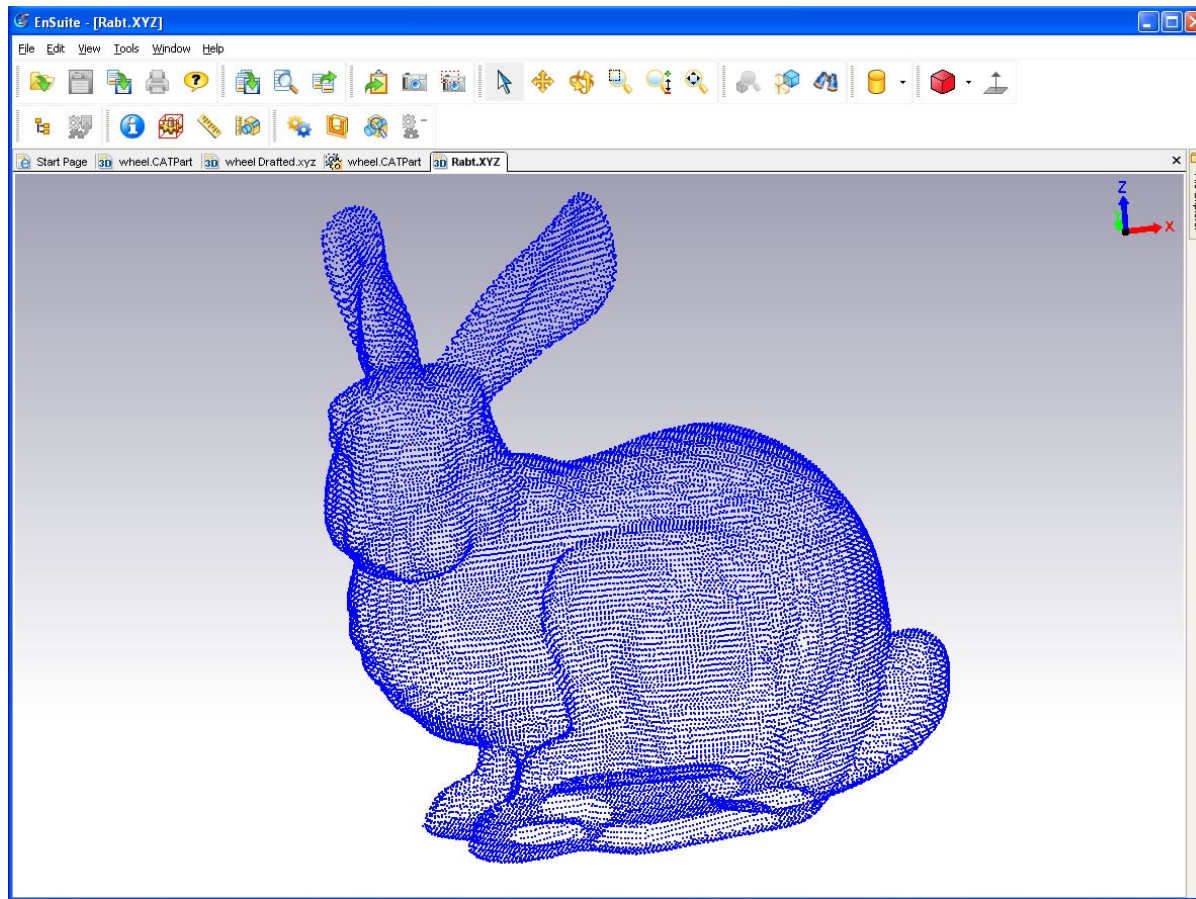


*Marching cubes:
Isosurface extraction*

http://en.wikipedia.org/wiki/Marching_cubes

Volumetric extraction

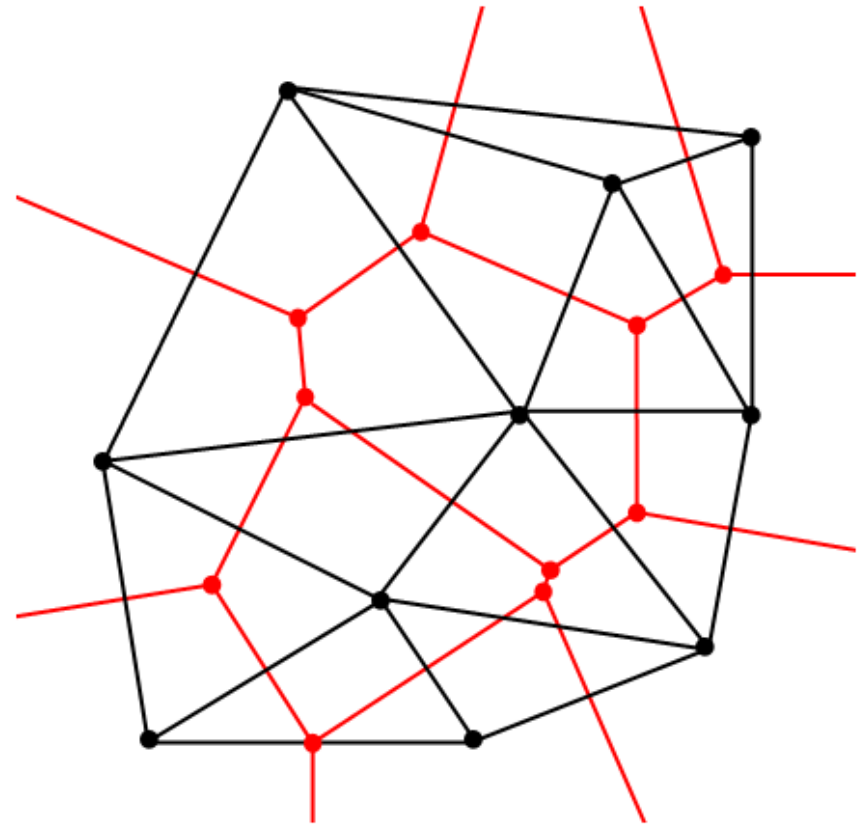
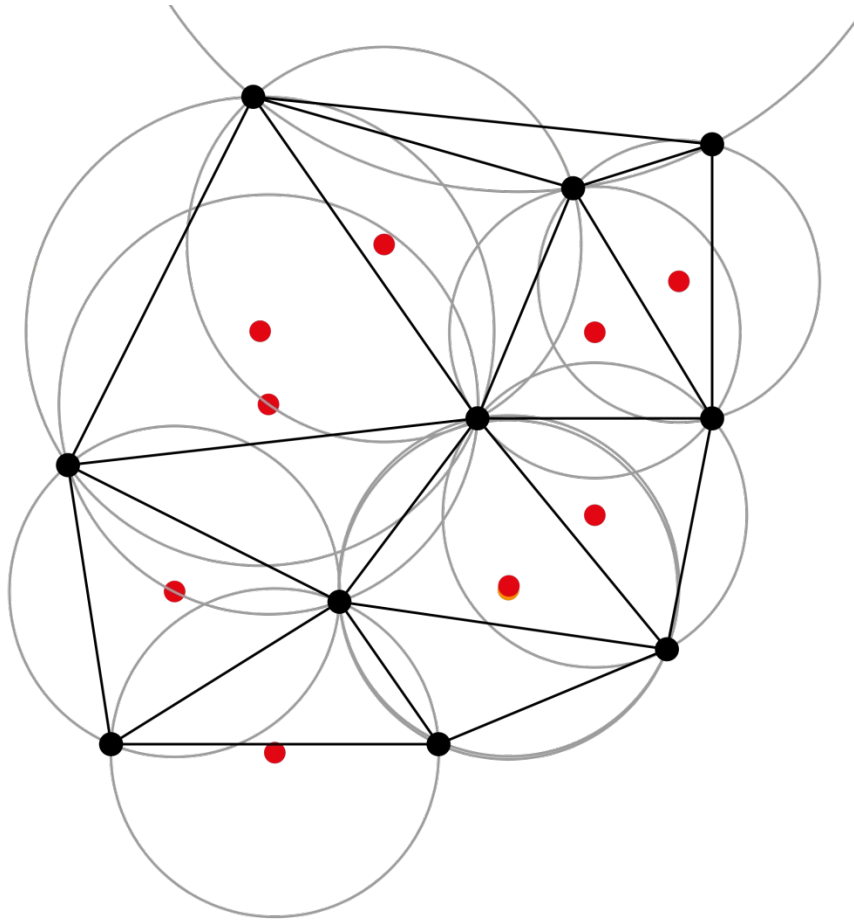
OBTAINING GEOMETRY



<http://www.engineeringspecifier.com/public/primages/pr1200.jpg>

Point clouds

DELAUNAY TRIANGULATION



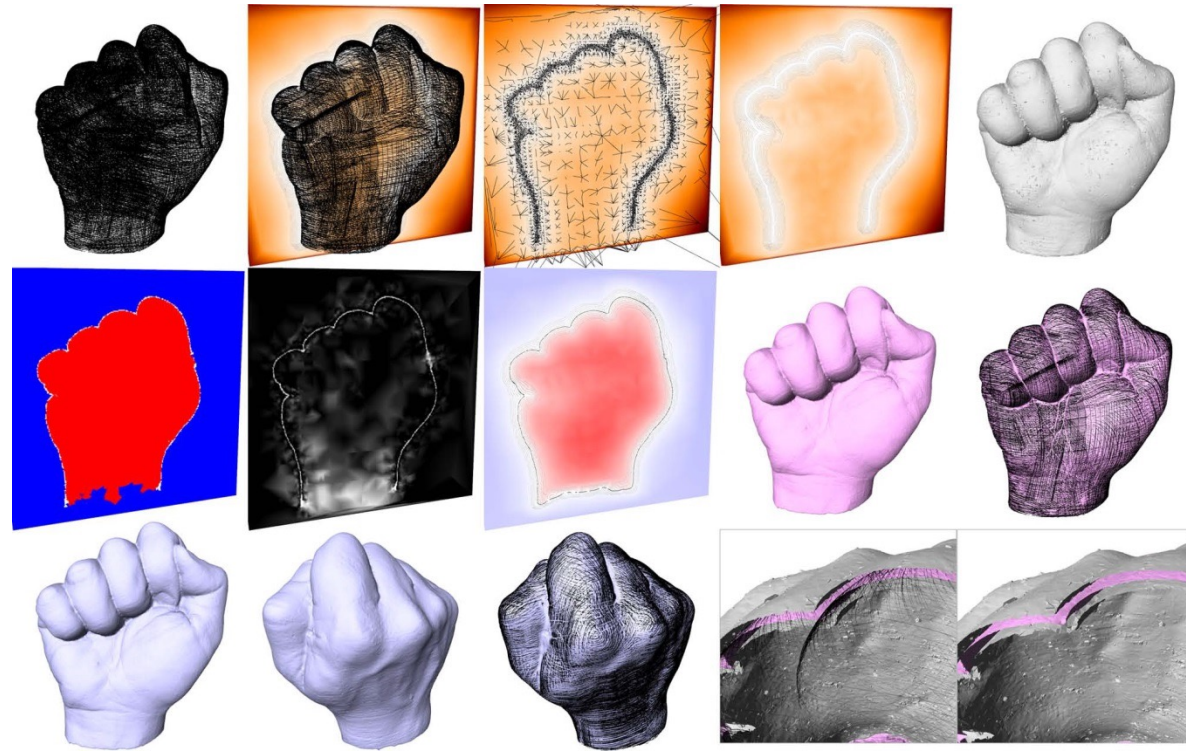
http://en.wikipedia.org/wiki/Delaunay_triangulation

Well-behaved dual mesh

STRATEGIES FOR SURFACE DELAUNAY

- ***Tangent plane***
Derive local triangulation from tangent projection
- ***Restricted Delaunay***
Usual Delaunay strategy but in smaller part of R^3
- ***Inside/outside labeling***
Find inside/outside labels for tetrahedra
- ***Empty balls***
Require existence of sphere around triangle with no other point

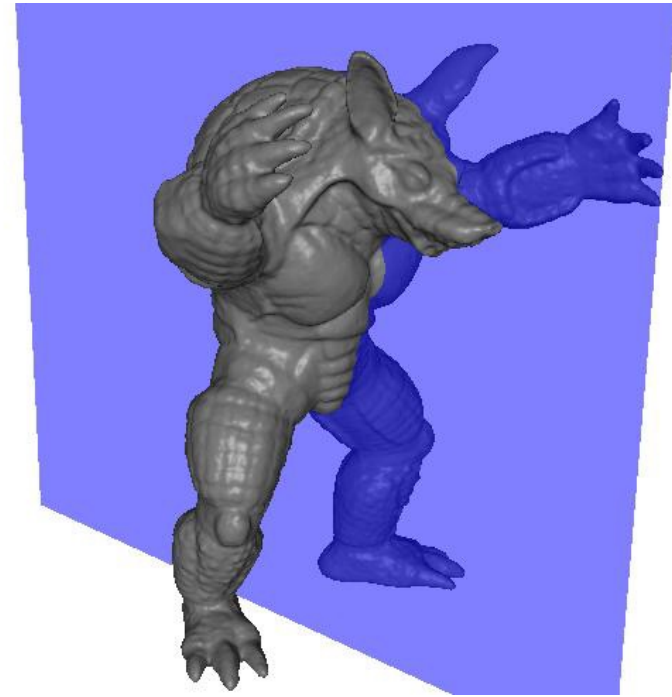
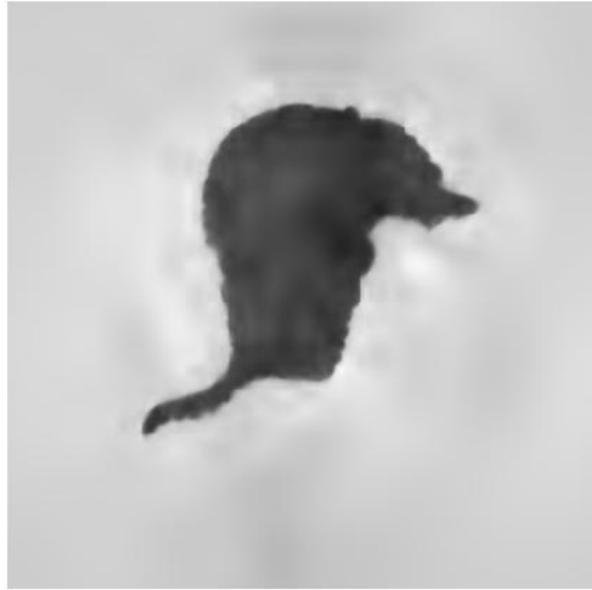
NOT THE ONLY MODEL



http://www.cs.umd.edu/class/spring2005/cmsc828v/papers/mpu_implicit.pdf <ftp://ftp-sop.inria.fr/geometrica/alliez/signing.pdf>

Implicit surfaces

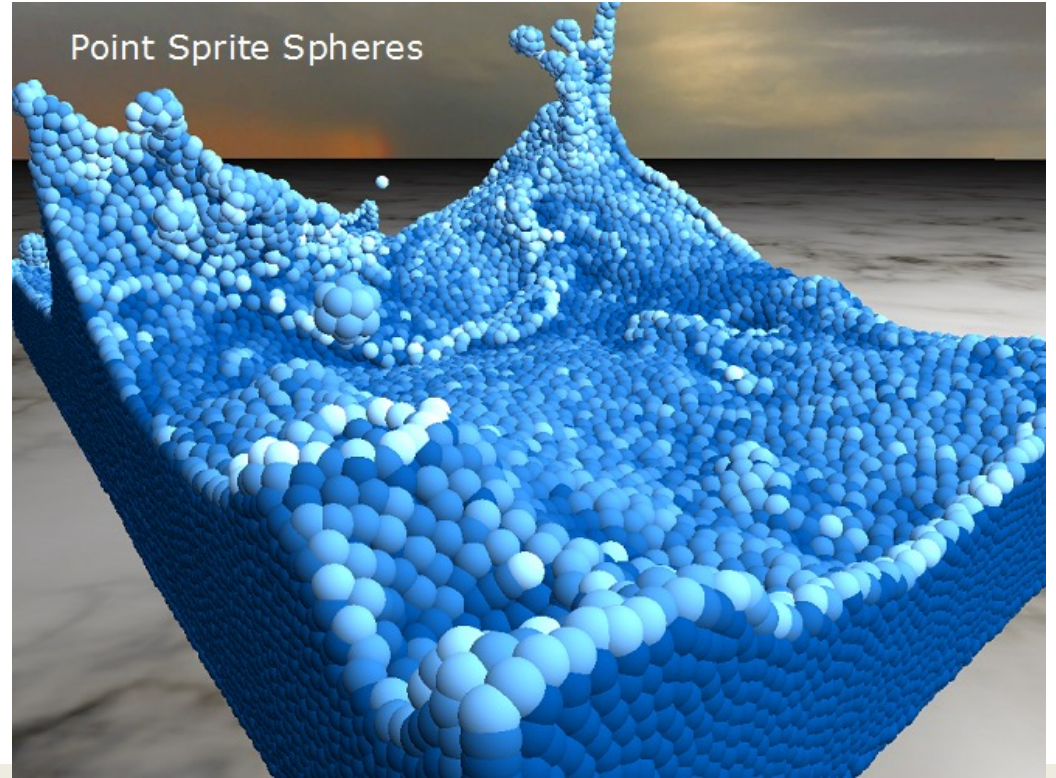
POISSON RECONSTRUCTION



*Poisson Surface Reconstruction
Kazhdan, Bolitho and Hoppe (SGP 2006)*

NOT THE ONLY MODEL

<http://www.itsartmag.com/features/cgfluids/>
<https://developer.nvidia.com/content/fluid-simulation-alice-madness-returns>



Smoothed-particle hydrodynamics

NEURAL NETWORK AS SURFACE REPRESENTATION?

DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation

Jeong Joon Park^{1,3†} Peter Florence^{2,3†} Julian Straub³ Richard Newcombe³ Steven Lovegrove³

¹University of Washington ²Massachusetts Institute of Technology ³Facebook Reality Labs

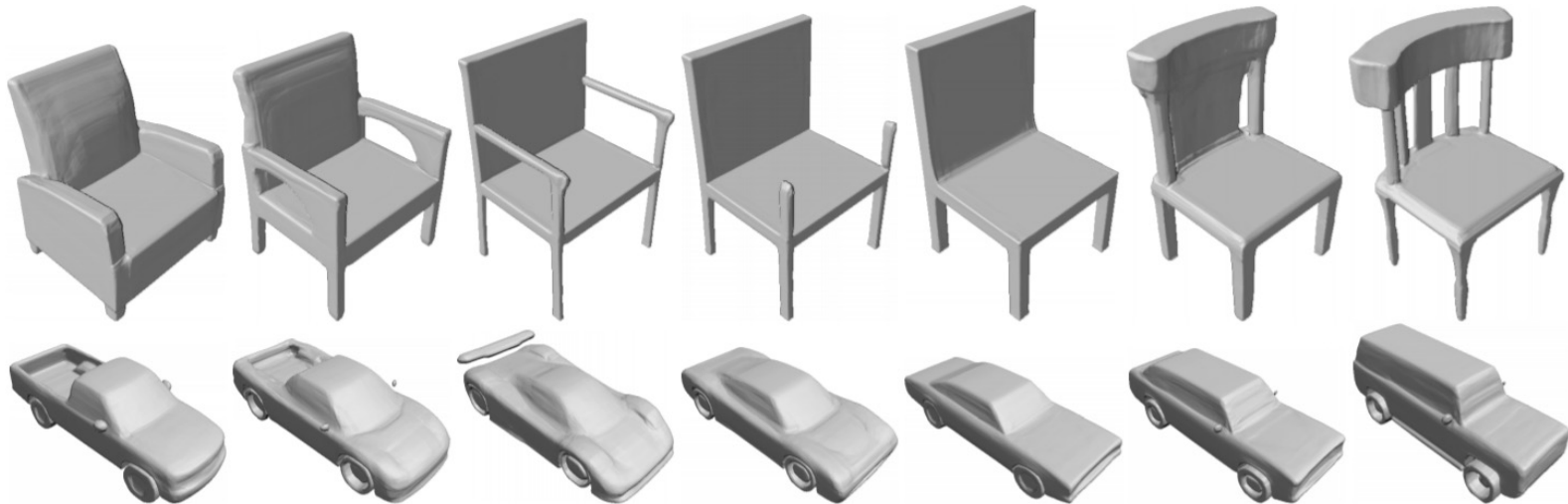


Figure 1: DeepSDF represents signed distance functions (SDFs) of shapes via latent code-conditioned feed-forward decoder networks. Above images are raycast renderings of DeepSDF interpolating between two shapes in the learned shape latent space. Best viewed digitally.