On The Use of Loop Subdivision Surfaces for Surrogate Geometry

Per-Olof Persson and Robert Haimes

Massachusetts Institute of Technology

Michael J. Aftosmis

NASA Ames Research Center

15th International Meshing Roundtable

September 19, 2006

Background

- Desire to reconstruct a smooth surface from a general triangulation
  - Analytic geometry lost (no CAD connection) – STLs, CT scans, etc
  - Light-weight substitute for CAD, simple and uniform view (vs. BRep)
- Applications
  - Surface remeshing – Need parametrization and evaluation capabilities
  - Inverse evaluations (find closest surface point, “snaps”) – Need second-derivatives for Newton iterations
  - Direct integration between geometry and solvers (Cut-cell methods, immersed boundary methods, etc)
  - Higher-order element construction
- Challenge: Define the smooth surface, provide required operations

Subdivision Surfaces

- Catmull and Clark (1978), Loop (1987)
- Widely used in the Computer Graphics community
- Generalization of splines to general meshes
- Represent smooth surface by a surface triangulation (“control-mesh”)
- C^2 almost everywhere, can specify creases and corners
- Fast operations:
  - Subdivision (creating a finer control mesh)
  - Computing normal directions (for lighting)
- Has been implemented in hardware (GPU)

Subdivision of Spline Curves

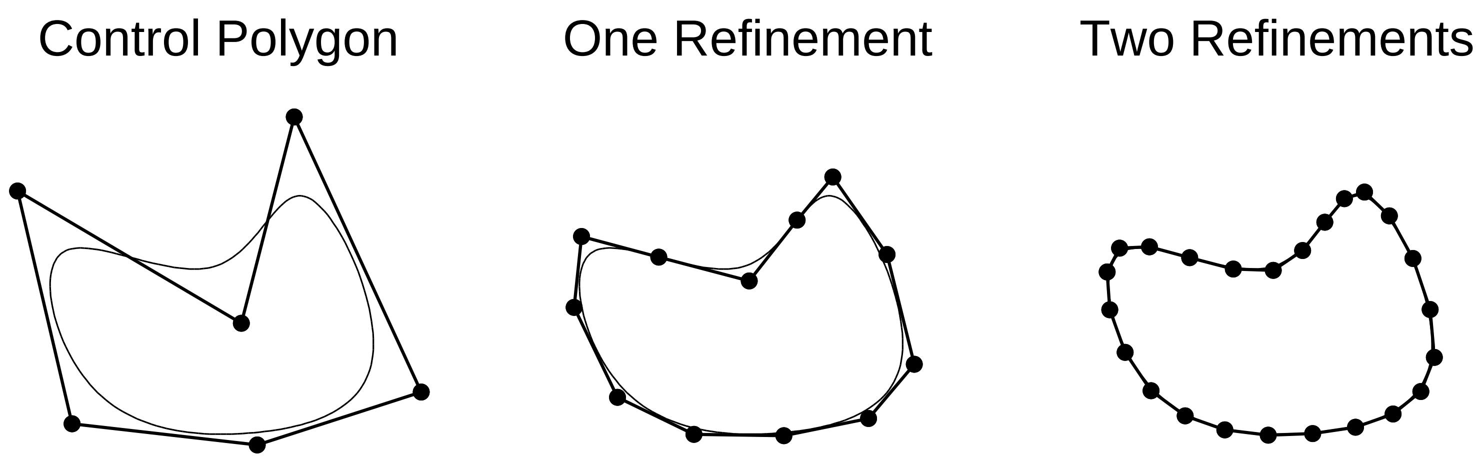
- Refinement of control polygon:

$$\text{Current nodes: } \boldsymbol{x}_{2i}^{j+1} = (\boldsymbol{x}_{i-1}^{j} + 6\boldsymbol{x}_{i}^{j} + \boldsymbol{x}_{i+1}^{j})/8$$

$$\text{New midpoints: } \boldsymbol{x}_{2i+1}^{j+1} = (\boldsymbol{x}_{i}^{j} + \boldsymbol{x}_{i+1}^{j})/2.$$

- Limiting node positions:

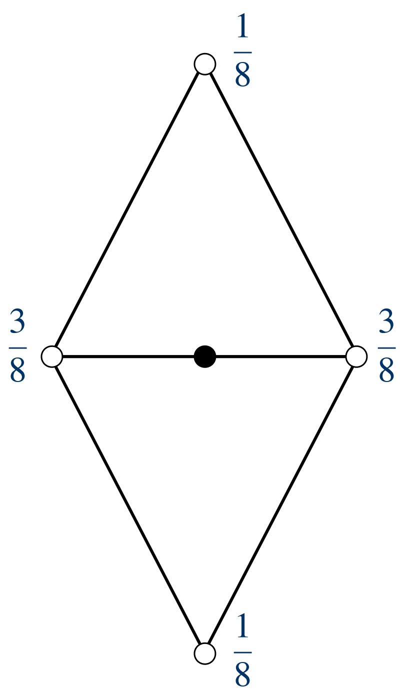
$$\text{Current nodes: } \boldsymbol{x}_{i}^{j,\infty} = (\boldsymbol{x}_{i-1}^{j} + 4\boldsymbol{x}_{i}^{j} + \boldsymbol{x}_{i+1}^{j})/6.$$



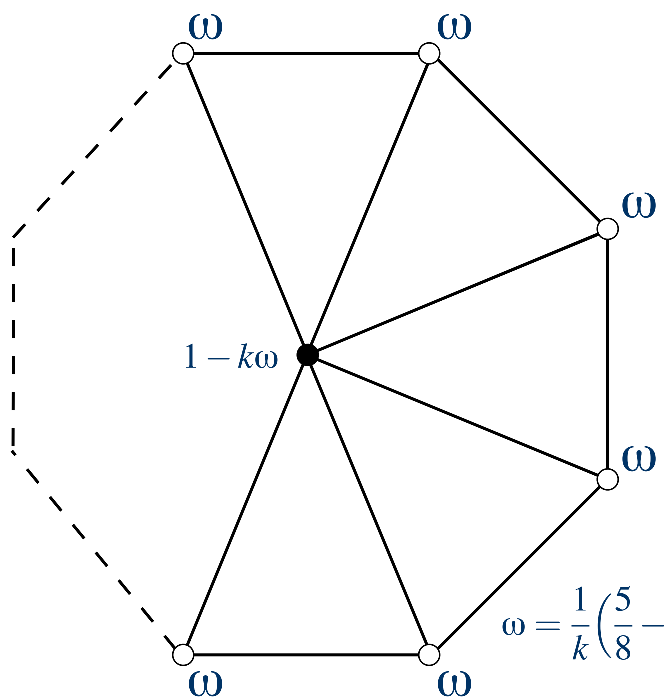
The Loop Subdivision Scheme

- Compute locations of new vertices as weighted average of original vertices
- Local neighborhood gives compact support
- Limiting node positions computed by similar scheme

Mid-edge insertion



Update of existing vertex location

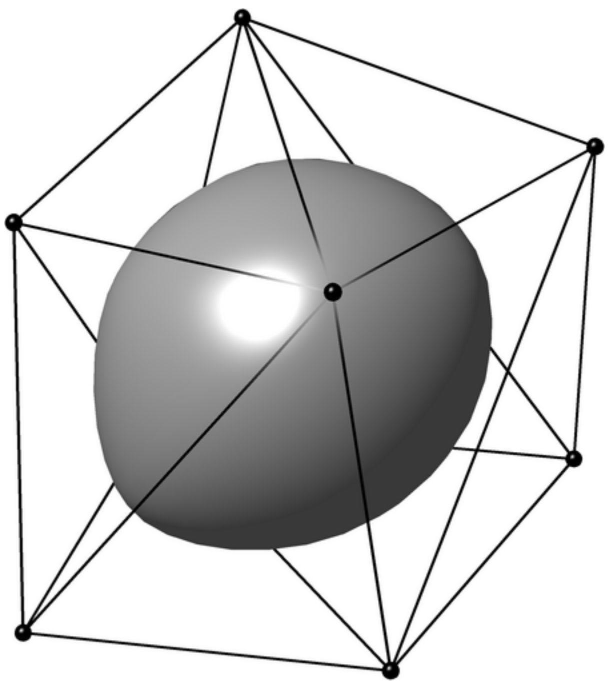


$$\omega = \frac{1}{k}\left(\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4}\cos\frac{2\pi}{k}\right)^2\right)$$

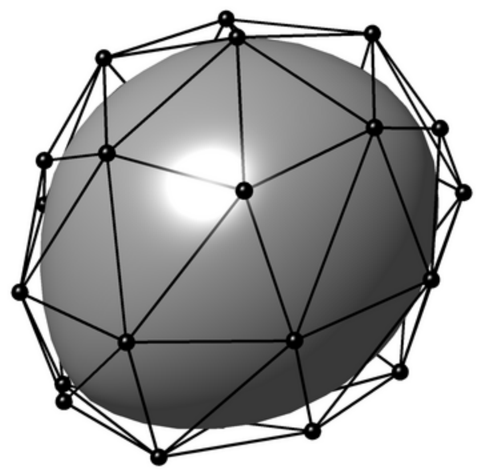
The Loop Subdivision Scheme

- In Computer Graphics, refine until sufficiently fine for visualization and normal directions (parametrization not required)
- Non-interpolating: Some shrinkage depending on smoothness
- Surface C^2 continuous everywhere except at irregular nodes

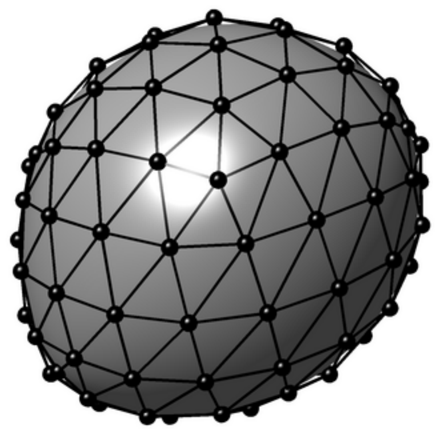
Control-Mesh



One Refinement



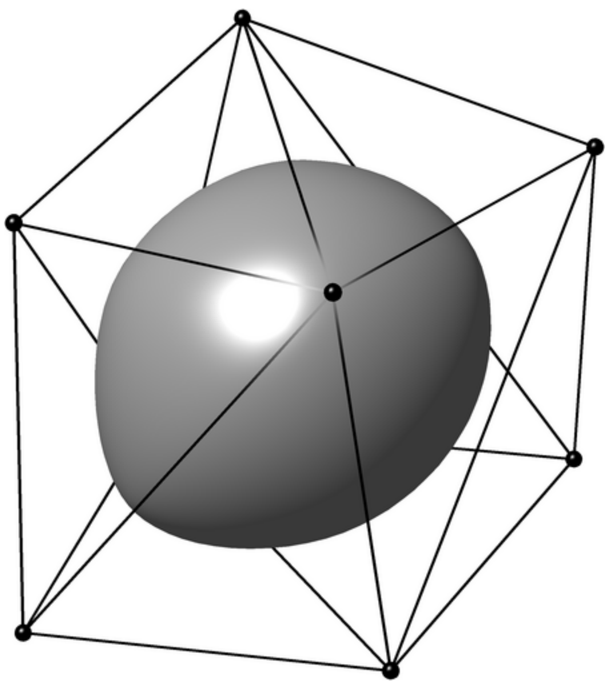
Two Refinements



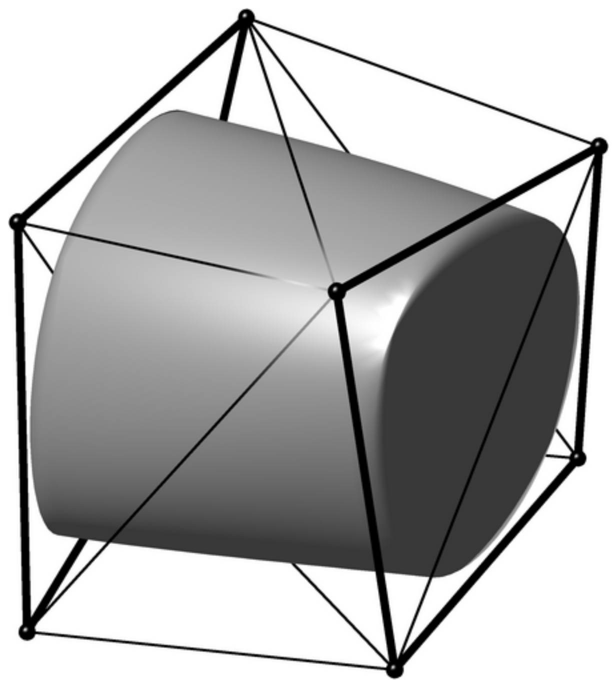
Hard Edges and Vertices

- Can specify triangle edge as “hard” (creases) – reduces to a spline curve
- For general input triangulations: Automatically detect hard edges/vertices based on angle criteria

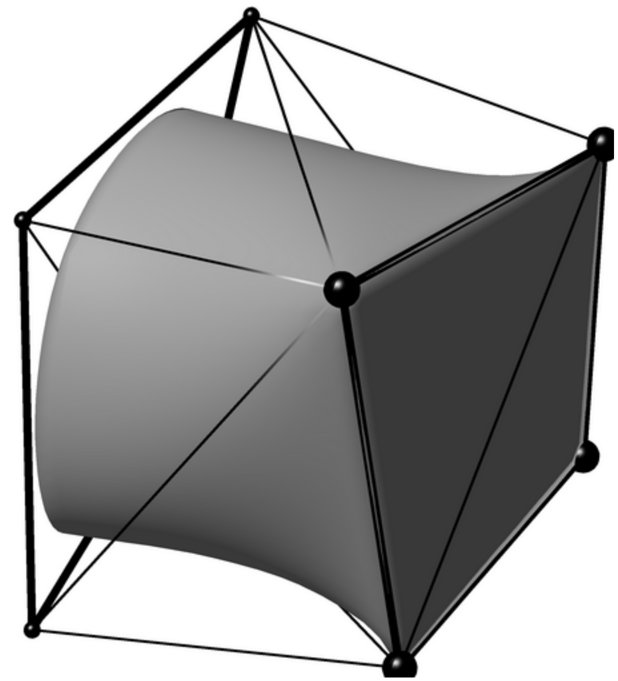
Soft Edges



Hard Edges



Hard Vertices



Evaluation and Parametrization

- Subdivision scheme can generate arbitrarily fine approximations of surface
- Our applications need general parametrization and evaluation capabilities
- Well-known expressions for spline curves, with four consecutive nodes $\boldsymbol{x}_{i-1}, \ldots, \boldsymbol{x}_{i+2}$, parameter value $t \in [0,1]$ between node i and $i+1$:

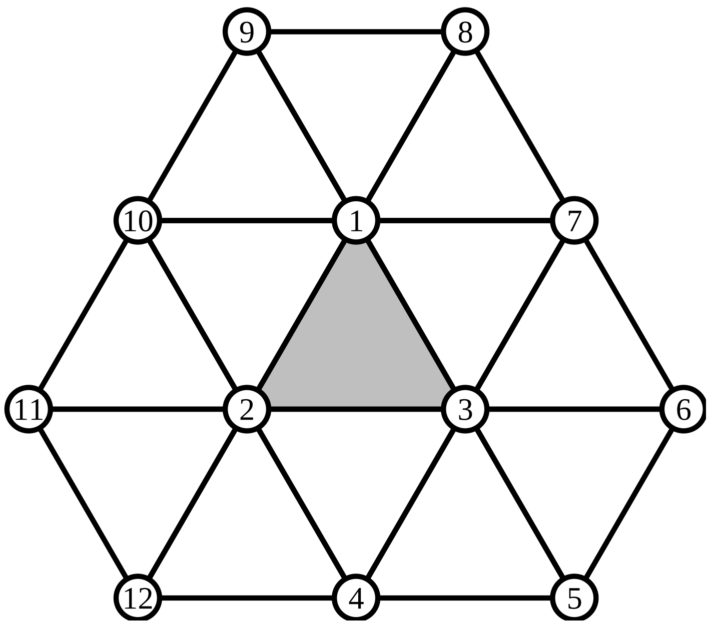
$$\boldsymbol{x}(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{i-1} \\ \boldsymbol{x}_i \\ \boldsymbol{x}_{i+1} \\ \boldsymbol{x}_{i+2} \end{bmatrix}$$

- Differentiate analytically for normals and curvatures

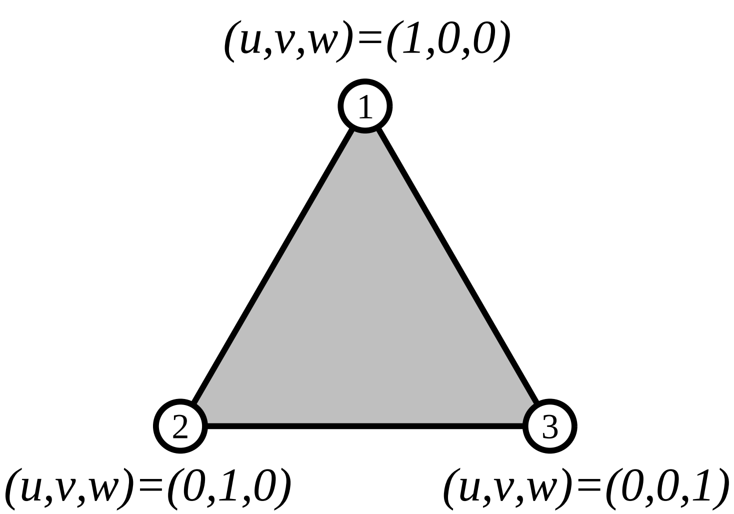
Parametrization – Regular Triangles

- In a regular triangle, the Loop scheme reduces to the common box splines
- Evaluate for arbitrary v, w by linear operation $\boldsymbol{x}(v,w) = \boldsymbol{x}\boldsymbol{\phi}\boldsymbol{c}(v,w)$, with local node locations $\boldsymbol{x}$ and monomial coefficients $\boldsymbol{c}(v,w)$
- Precomputed constant matrix $\boldsymbol{\phi}$ (see paper for details)

Local Node Numbering

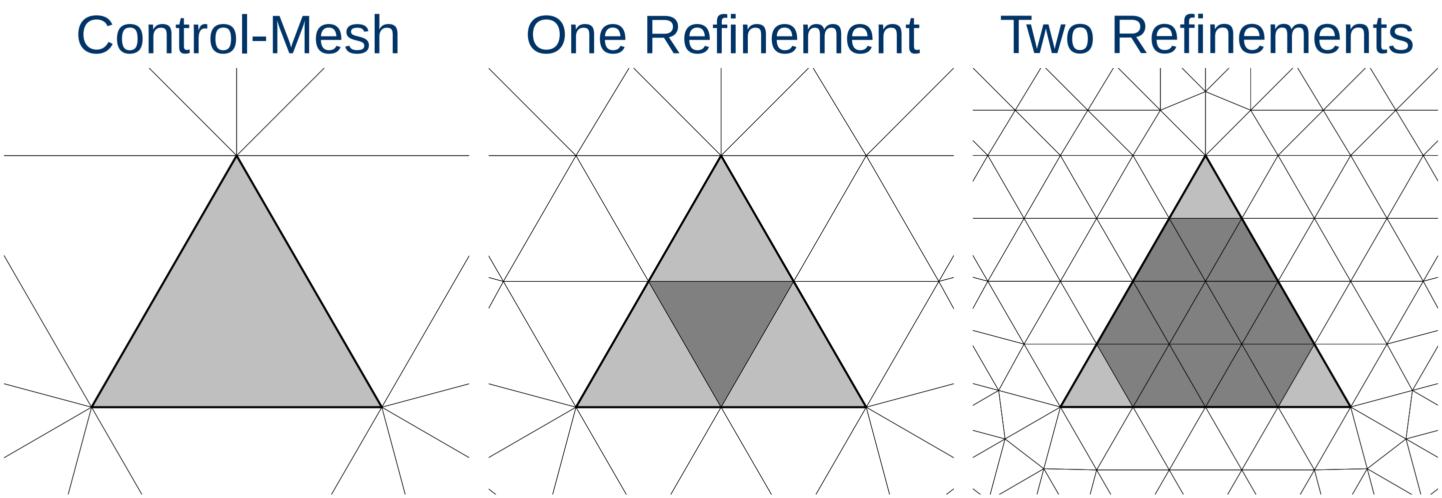


Local Coordinates



Parametrization – Irregular Triangles

- Main idea [Stam 1998]: Subdivision introduces new triangles with regular neighborhoods, which can be evaluated as box splines
- Close to irregular nodes, large number of refinement might be required
  - Stam solved this using the eigen-decomposition of refinement matrix
  - Simpler hack: Nudge requested position to center after 25 subdivisions
- Implementation: Local representation of triangle neighborhood
  - Too expensive to refine entire mesh



The Evaluation Algorithm

function $[\boldsymbol{x}, \boldsymbol{dx}, \boldsymbol{ddx}] = \mathbf{loopeval}(t, v, w, depth)$

if t regular

Evaluate surface location and derivatives using box-spline expressions

else

Subdivide t and all triangles sharing its nodes

if $depth \leq 25$

$t' =$ new triangle covering v, w

$v', w' =$ new local coordinates in t'

else

$t' =$ center subtriangle

$v', w' =$ new local coordinates (on edge of subtriangle)

end if

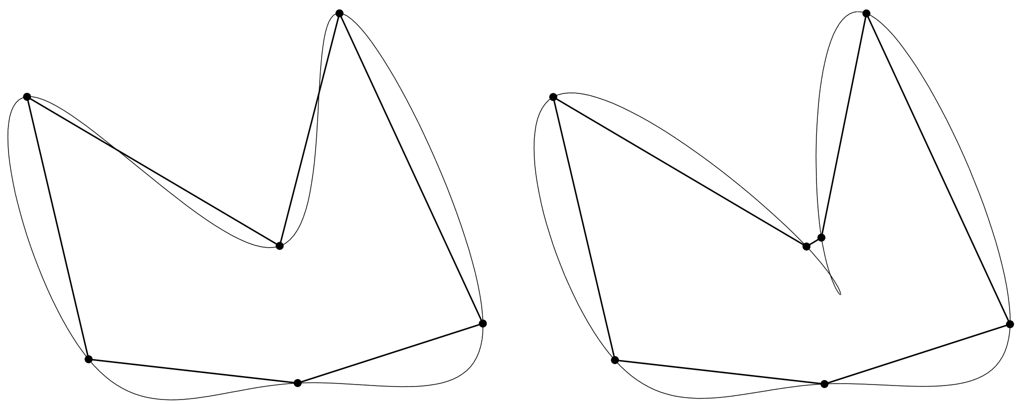
$[\boldsymbol{x}', \boldsymbol{dx}', \boldsymbol{ddx}'] = \mathbf{loopeval}(t', v', w', depth + 1)$

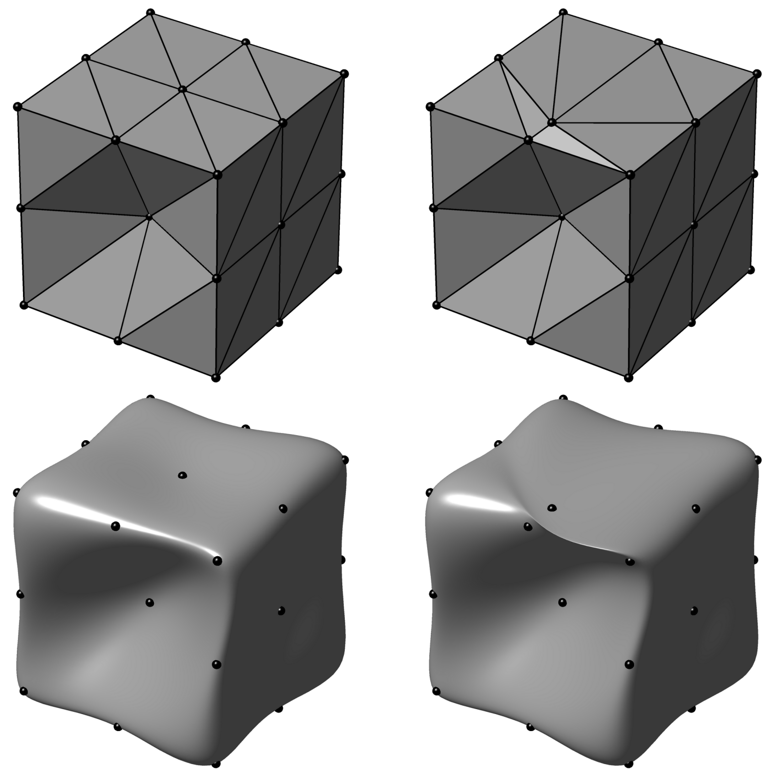
Inverse mapping: $\boldsymbol{x} = \boldsymbol{x}',\ \boldsymbol{dx} = \pm\boldsymbol{dx}'/2,\ \boldsymbol{ddx} = \boldsymbol{ddx}'/4$

end if

Creating an Interpolating Scheme

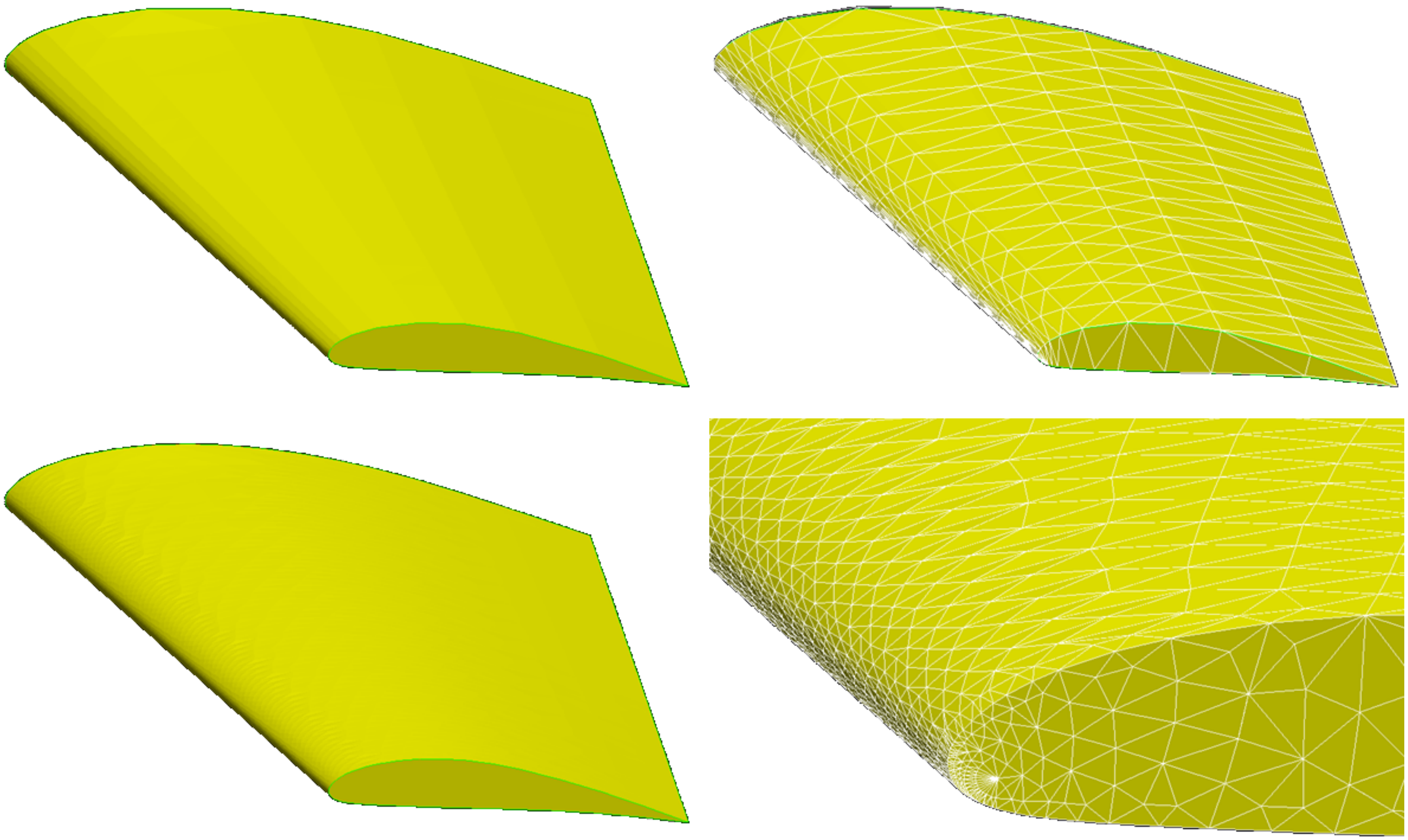
- The interpolating subdivision schemes produce less smooth surfaces
- Alternative: Solve for new control-mesh mesh nodes in $S^{\infty}(\boldsymbol{x}_{\mathrm{interp}}) = \boldsymbol{x}$
  - Well-conditioned, solve using a few Krylov iterations
  - Good for uniform smooth control-meshes, but might produce cusps



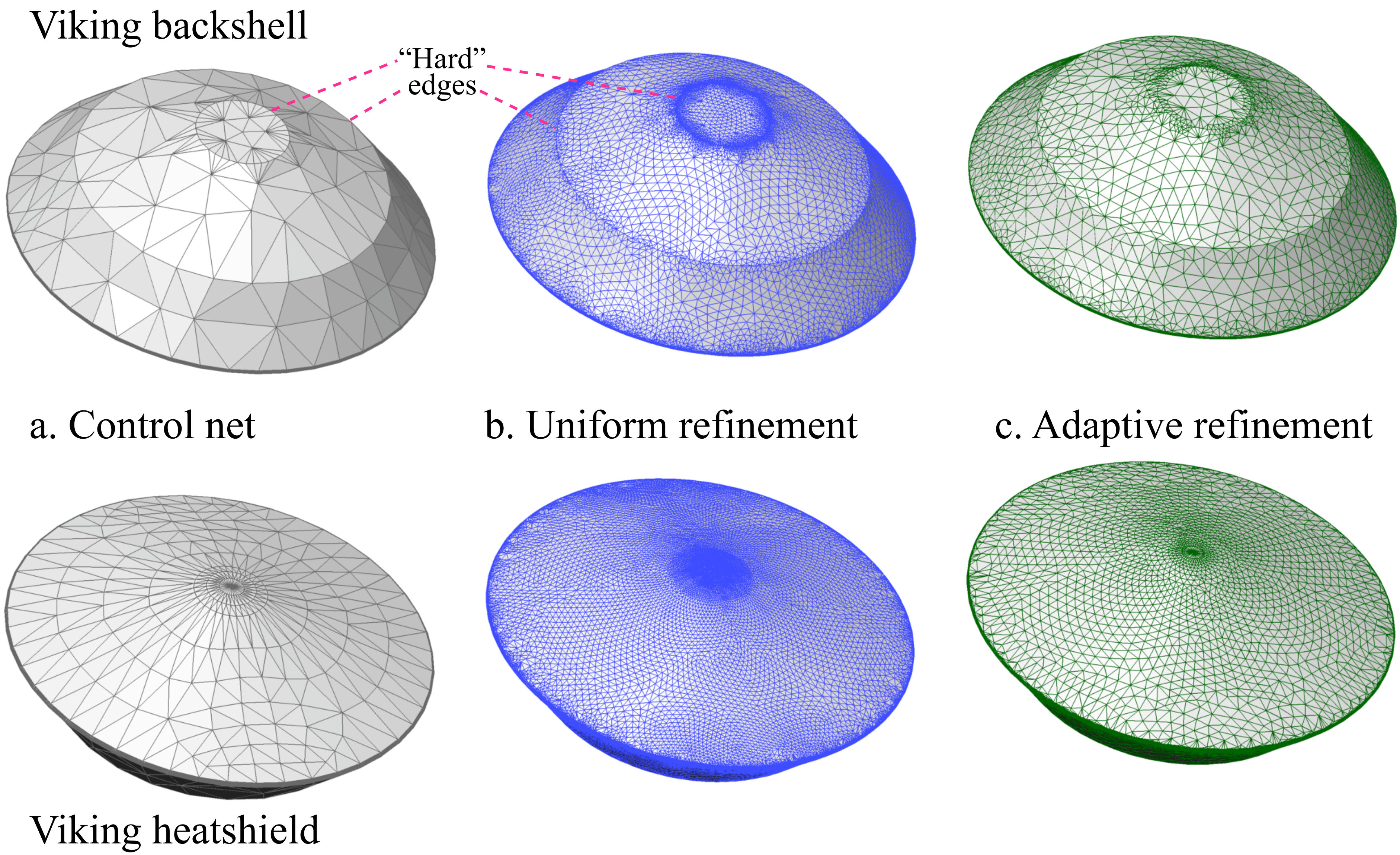


Example: Flap

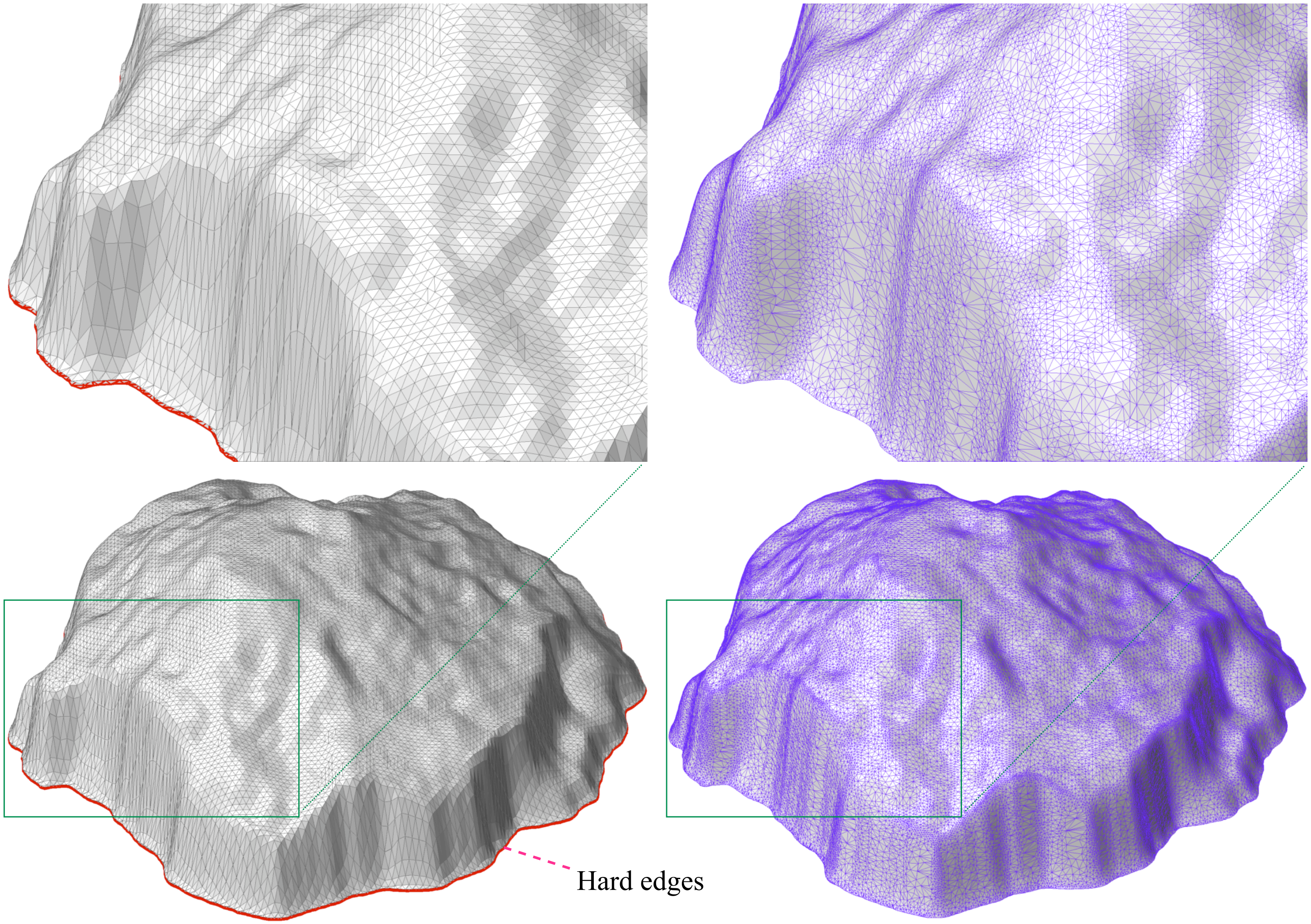
- Simple mesh refinement for real-world geometries provided in STL format
- Demonstration of arbitrary parameter evaluation, not high-quality meshing



Example: Viking Spacecraft Aeroshell



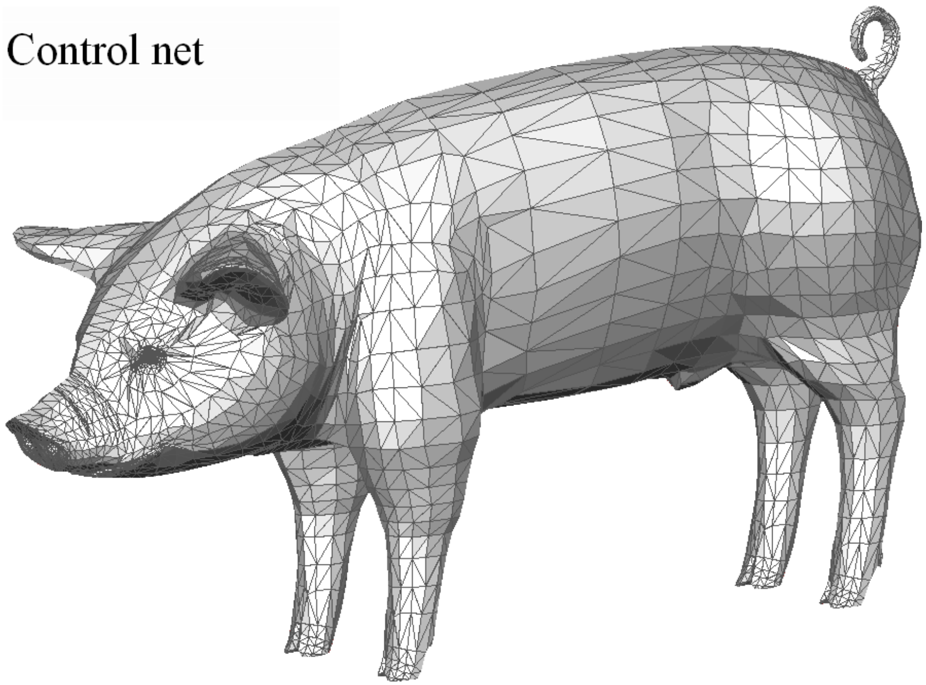
Example: Scanned Foam Piece

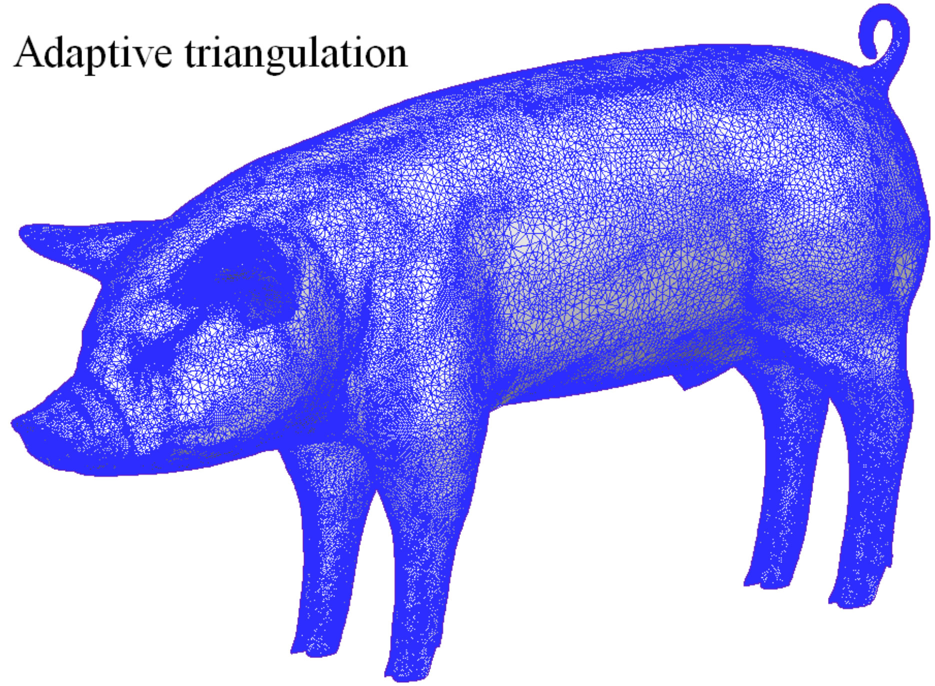


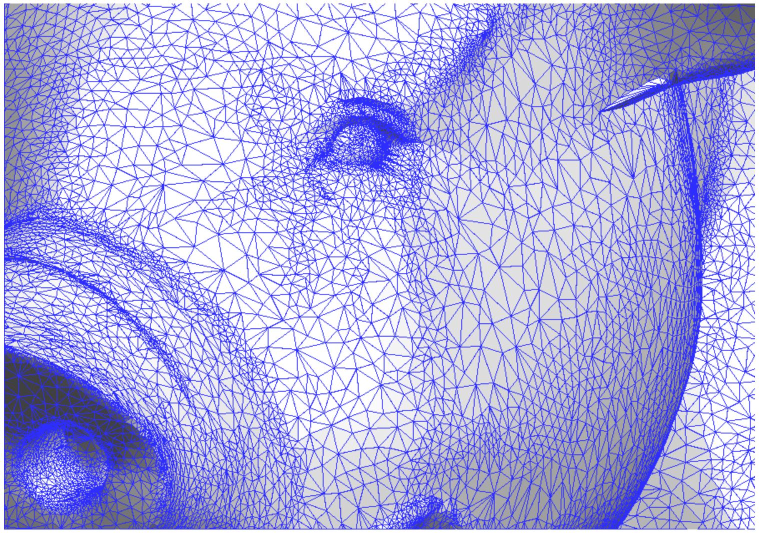
Control net

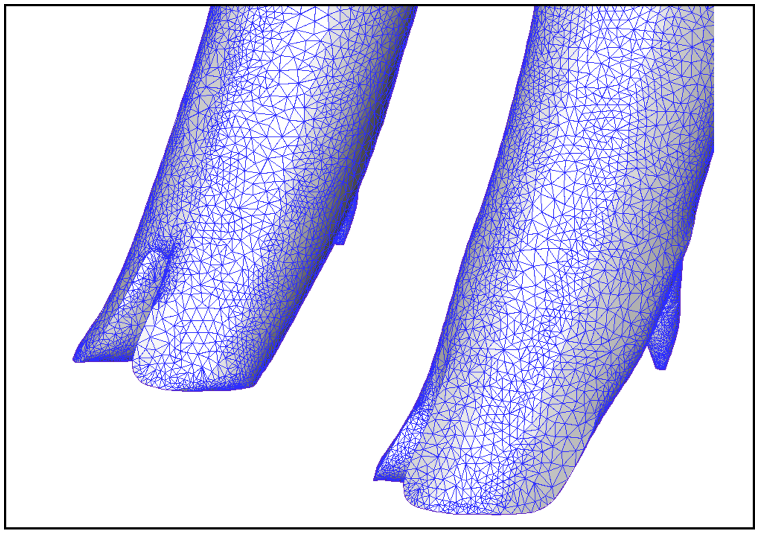
Adaptive triangulation

Example: Pig









Conclusions

- Use subdivision surfaces as surrogate for real CAD geometry
- Simple implementation, uniform view
- Remeshing, adaptation, inverse evaluations, etc
- TURIN – the Tetrahedral Unstructured Remeshing INterface
  - Contact Bob Haimes (haimes@mit.edu) for more information