Automated Discretization of Digital Curves through Local or Global Constrained Optimization

David O. McLaurin, Ph.D.

Presentation Outline

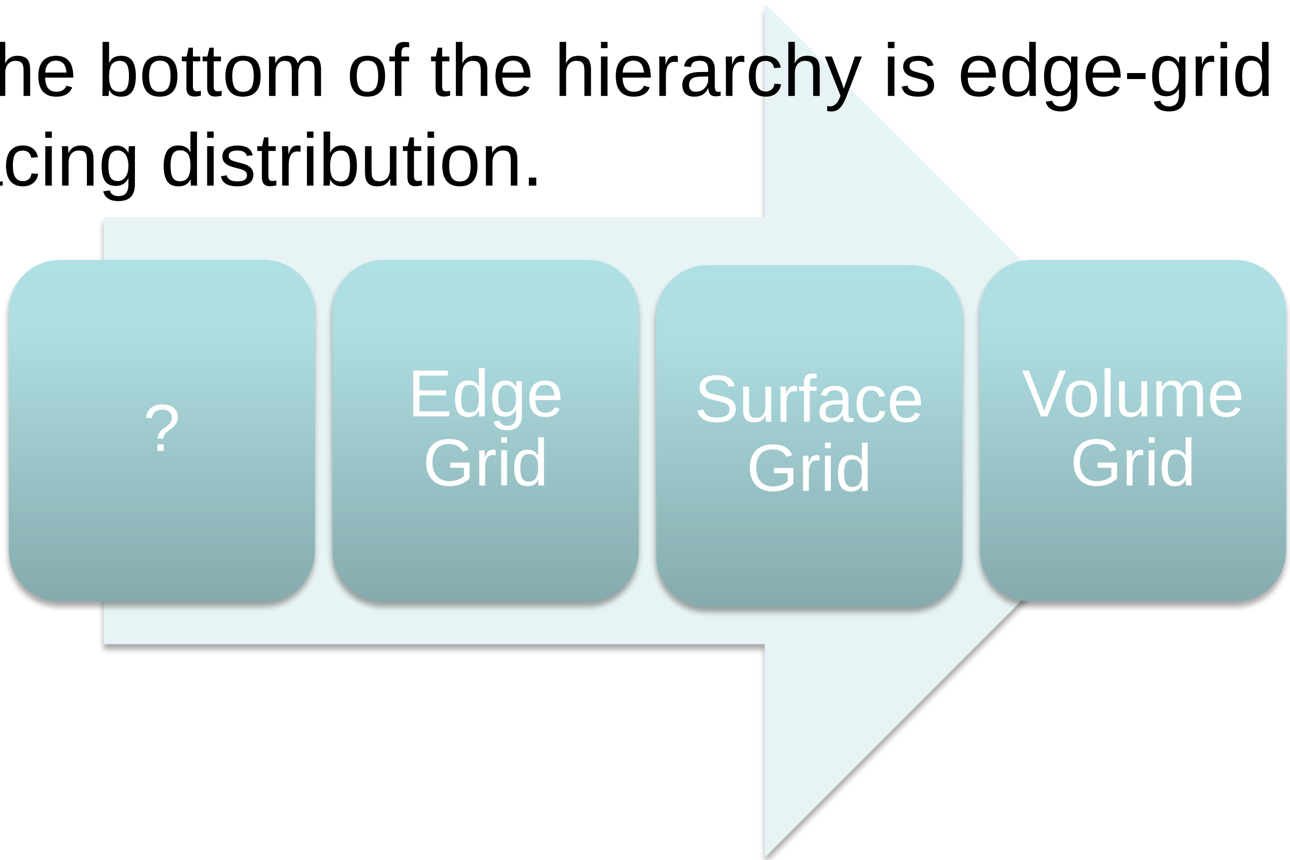
- Motivation / Problem Description
- Related Solutions
- Proposed Solution: Curve Discretization through Optimization
- Application: Implementation of Proposed Solution in Existing Mesh Generator
- Results / Conclusions
- Questions

Motivation / Problem Description

- Computational analysis and design has a fundamental role in research, development, and manufacturing.
- Discrete model accuracy heavily influences simulation accuracy.
- Making accurate discrete models takes a significant amount of time.
- The aim of this work it to reduce the time required to prepare a model for volume grid generation.

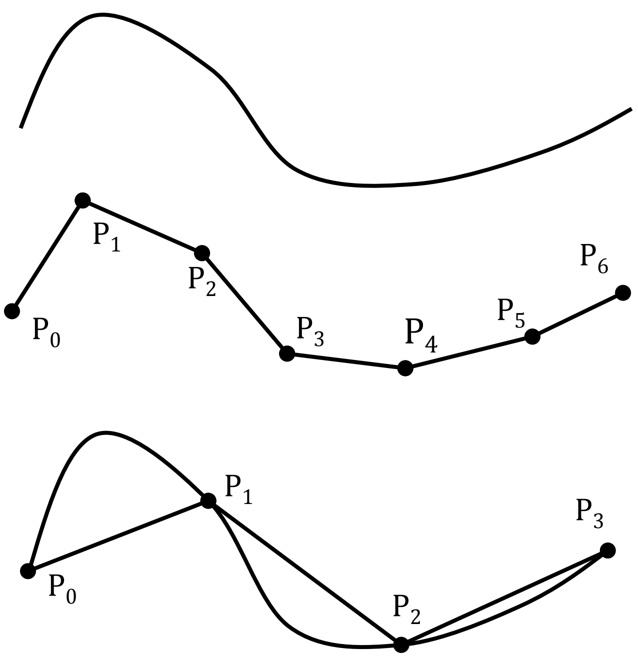
Mesh Generation Hierarchy

- Each of these processes is highly automated given a definition from below in the hierarchy.
- At the bottom of the hierarchy is edge-grid point spacing distribution.



Arc-Length Deficit—Definition

- Consider the two discretizations of the curve: the upper one is “better” than the lower one because the length of the combined segments more closely approximates the length of the curve.
- Traditional grid generators create high-quality elements.
- Quality can always be improved, but, first, the geometry must be represented accurately.
- Arc-Length Deficit: difference between length of segments in edge-grid and arc-length of curve.



Arc-Length Deficit—Alternate Methods

- Feature based grid generation, most often called "adaptive mesh refinement", presupposes that the implemented scheme will accurately represent the geometry once the desired feature is captured.
- Curvature based grid refinement is one of the most popular choices—but is not general enough to be useful for most applications.

Arc-Length Deficit—Issues with Feature Based Grid Refinement

- Feature based methods are geometry-representation specific.
- Usually require derivative information that might not be available everywhere—or is difficult/expensive to evaluate.

Arc-Length Deficit—Discretization by Global Optimization

- Given a curve in R^3, C with a normalized parameterization variable, $0 \le u \le 1$, $C(u)$: each u will correspond to a point in R^3.
- A number of interior points is chosen, n_i. Such that there are $(n_i+2)=N$ total points (including the end points), $(u_0,u_1,u_2,\ldots u_N)$.
- The topology of the grid is fixed in the form of constraints: such that for each point on the grid, $(u_{i-1}<u_i)$ and $(u_i<u_{i+1})$.
- Optional: Minimum (ε) and maximum (μ) edge length for the discretization could be enforced through more constraints: $\varepsilon \le (C(u_{i-1})-C(u_i))$ *and* $(C(u_{i-1})-C(u_i)) \le \mu$.

Arc-Length Deficit—Discretization by Global Optimization (continued)

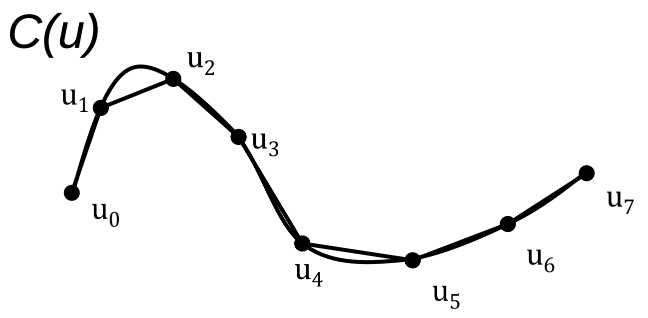
- The optimization function is then the sum of the length of the all of the segments:
- For j=0…(N-1), the segment S_j is defined by the edge between $C(u_j),C(u_{j+1})$. Therefore the optimization function is defined as:

$$O(u_1, u_2, u_3, \ldots u_{N-1}) = -\sum_{j=0}^{N-1} length(S_j)$$

- The optimal discretization for n_i interior points can then be found be finding $(u_1,u_2,u_3,\ldots u_{N-1})$ such that the optimization function in minimized.

Arc-Length Deficit—Discretization by Global Optimization (continued)

Example parameterized curve:



n_i=6 N=8 $S_0=\{C(u_0),C(u_1)\}$
$S_1=\{C(u_1),C(u_2)\}$
$S_2=\{C(u_2),C(u_3)\}$
$S_3=\{C(u_3),C(u_4)\}$
$S_4=\{C(u_4),C(u_5)\}$
$S_5=\{C(u_5),C(u_6)\}$
$S_6=\{C(u_6),C(u_7)\}$

With a length function, L(S), that calculates the length of a segment the optimization function is:

$$O(u1,u2,u3,u4,u5,u6) = -\left[L(S_0)+L(S_1)+L(S_2)+L(S_3)+L(S_4)+L(S_5)+L(S_6)\right]$$

Subject to the constraints for topology and (optionally) minimum and maximum edge lengths.

Arc-Length Deficit—Problems with Global Optimization

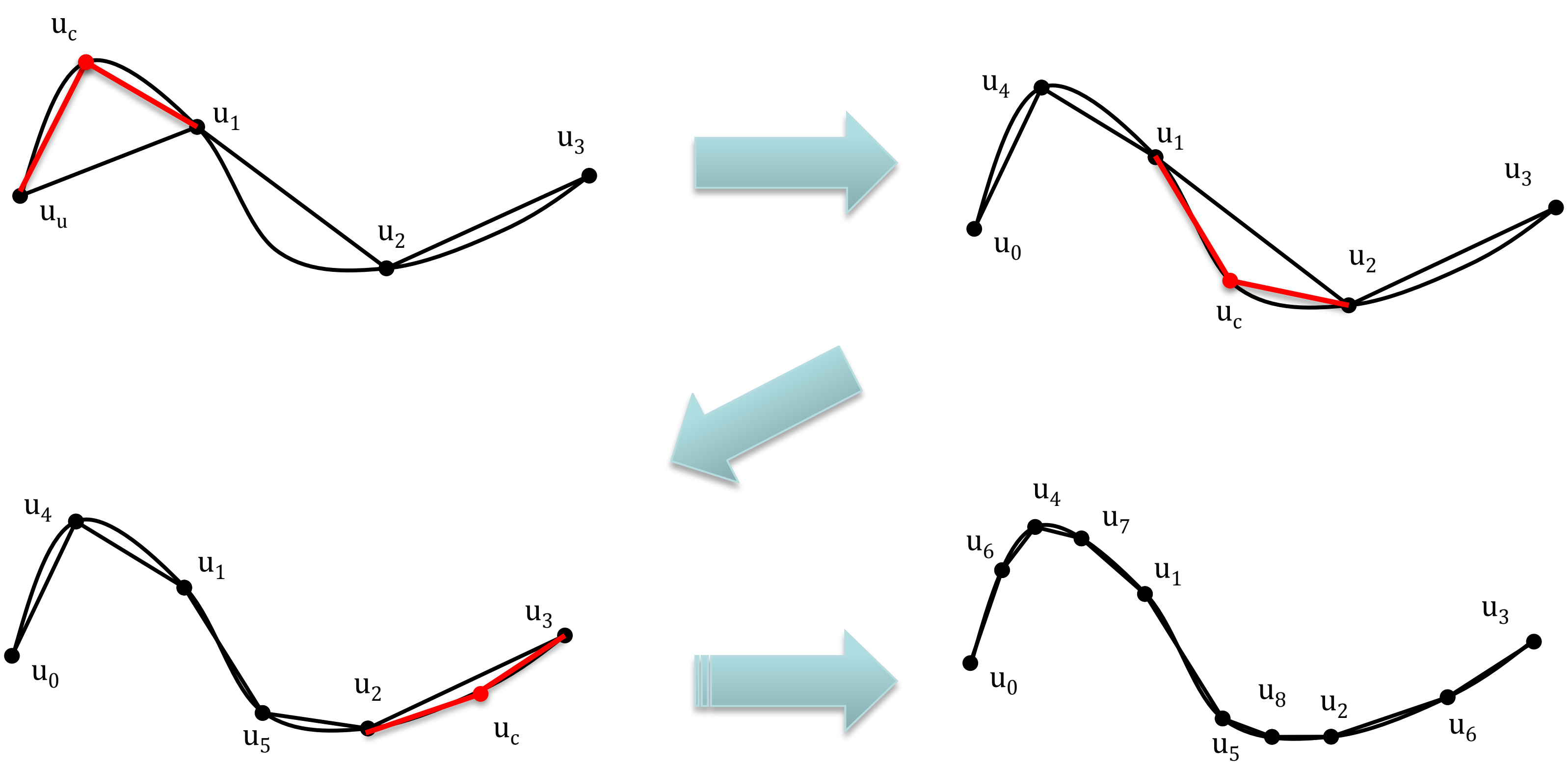
- Global optimization can be computationally expensive for large grids.
- How to pick the number of interior points?
- A (computationally) cheaper option is needed because the purpose is to accelerate the process of geometry preparation—not relocate the bottleneck to the bottom of the grid hierarchy.

Arc-Length Deficit—Discretization by Local Optimization

- Instead, use local optimization to calculate *an* optimal solution.
- This takes advantage of the *optimal substructure* of this problem.
- By obtaining an optimal solution with $n_i=1$ for each edge, every edge will be optimally represented.
- The combination of the locally optimal edges represents an optimal solution for the entire curve.

Optimal Grid Generation—Locally Justified, Optimal Edge-Grid

- Below is a locally justified, optimal edge grid generation process



Optimal Edge Grid Generation Algorithm

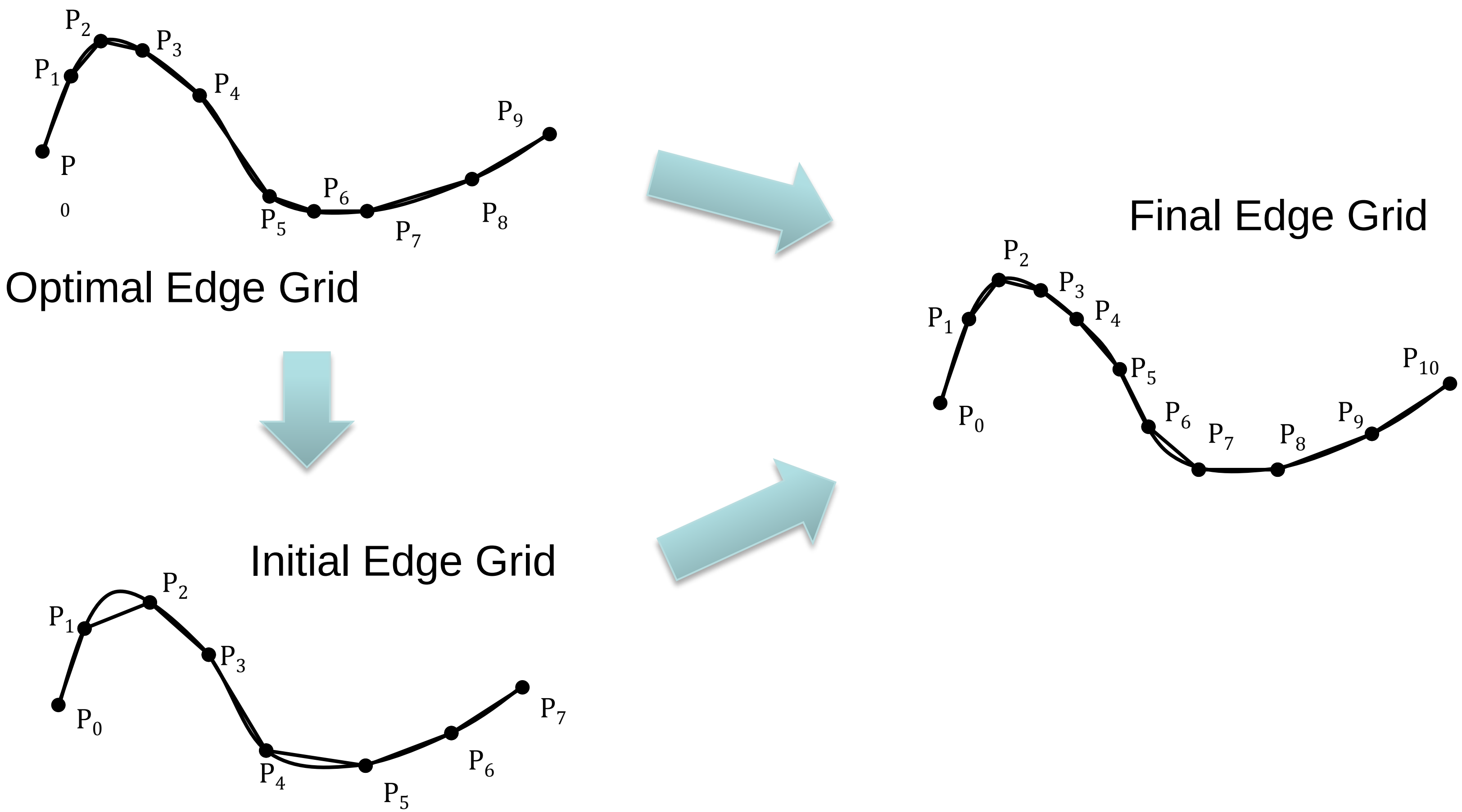
- For each curve:
  - Define *edge-grid* as a queue of line segments (*edges*), initially only containing the line segment connecting the two end points of the curve.
  - Define *optimal-criteria*: percent change in arc length, minimum and maximum edge length.
  - Breadth-first, simulated recursion is used to insert optimal points into *edge-grid*.
  - While queue is not empty:
    - Pop edge from queue
      - Calculate point on interior of edge on curve that minimized the local arc length deficit.
      - Determine if point should be placed based on *optimal-criteria*
      - If yes:
        - Push the two new edges into queue
      - If no:
        - Do nothing

Algorithm for Implementing Optimal Edge Grid into Existing Grid Generator

- For each *edge-grid*:
  - Use end-point point-spacing values from *optimal edge-grid* to generate *initial edge-grid*
  - Define *evolving edge-grid* as *initial edge-grid*
  - Set user-defined *Deviation Factor*
  - Loop over intervals in *evolving edge-grid*
    - Find interval with largest *Deviation Factor*
      - Compare Local *Deviation Factor* to user-defined *Deviation Factor*
        - Insert point-spacing source on the interior of the curve that matches point-spacing value for the *optimal edge-grid* for that interval
        - Regenerate edge-grid with included point-spacing source
    - Repeat if any point-spacing sources have been added

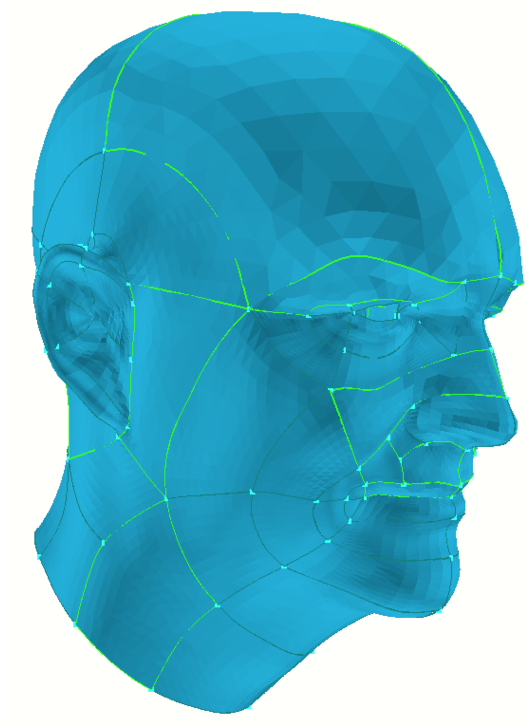
Implementation in Real-World Grid Generator

- SolidMesh Edge-Grid Generation

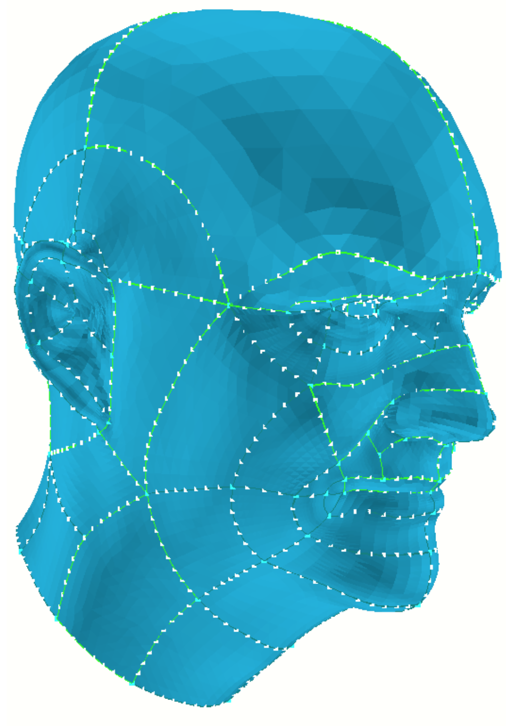


Results

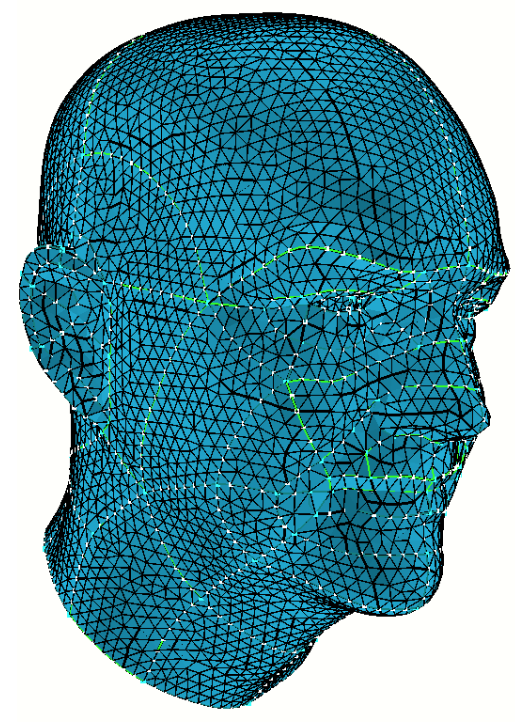
- Male Head, from CAD model with uniform point distribution



CAD



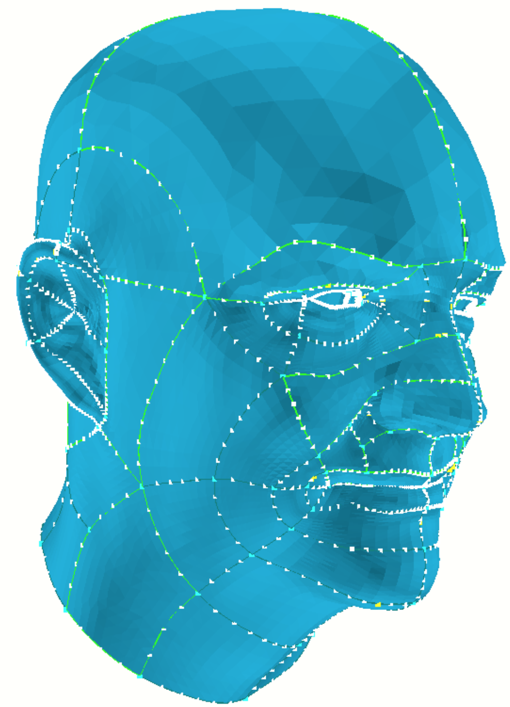
Edge Grid



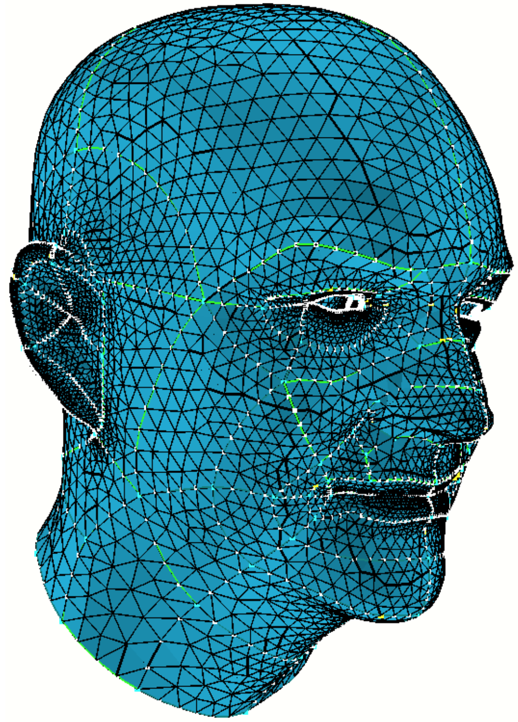
Surface Grid

Results

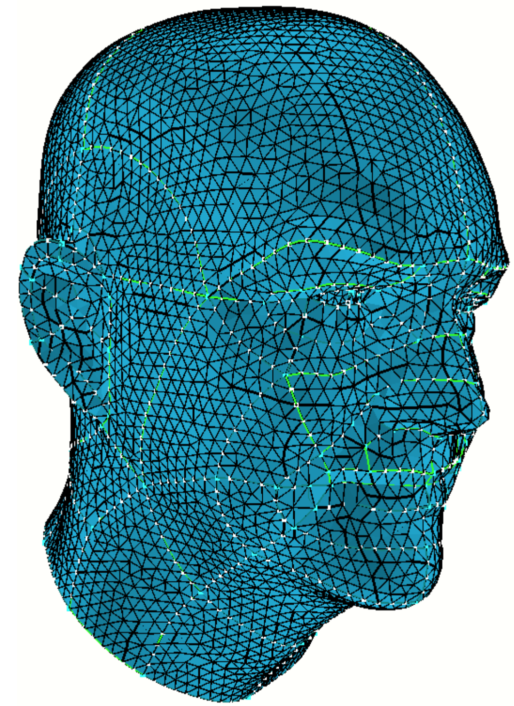
- Male Head, from CAD model with optimally-generated point distribution (compare to



Edge Grid



Surface Grid from Optimal Edge Grid

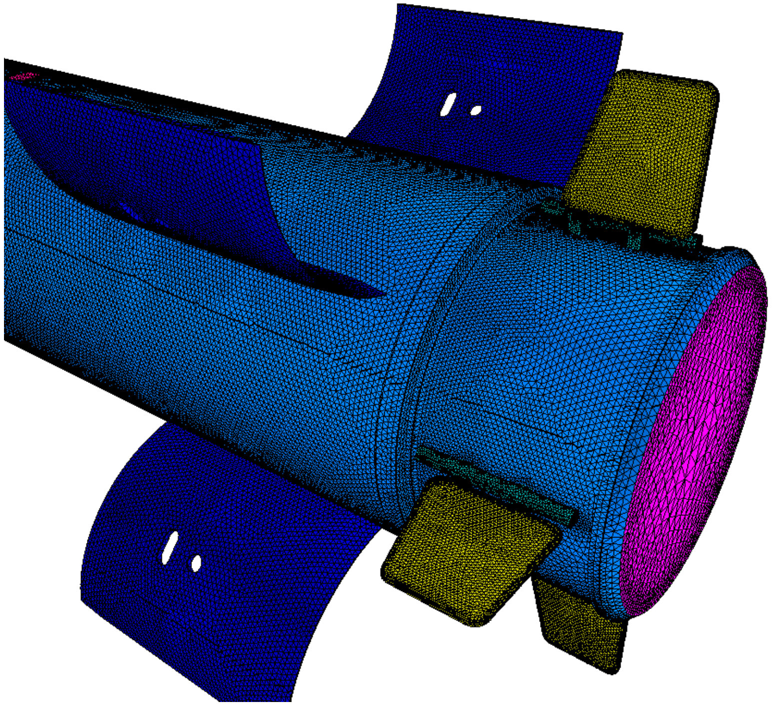


Uniform Surface Grid

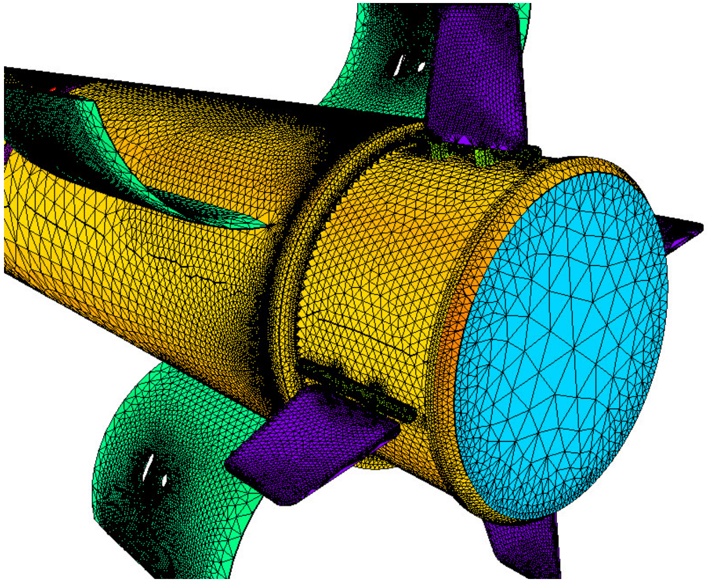
Results

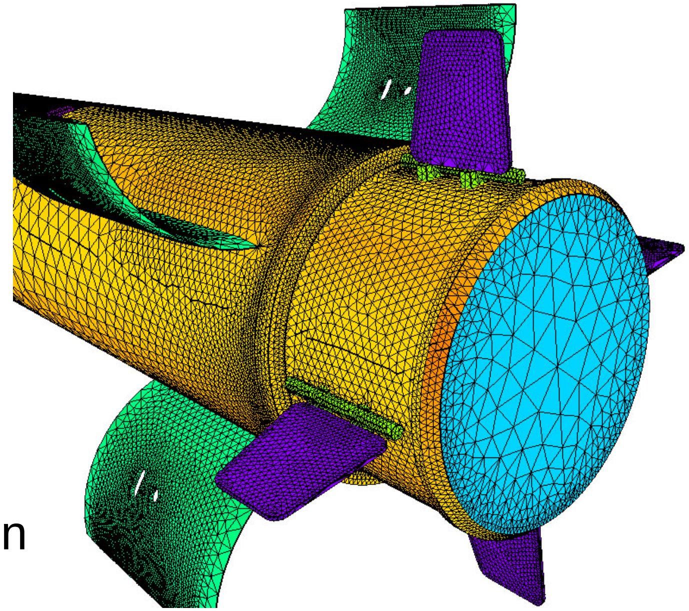
❑ Missile Tail Section

Uniform Surface Grid



Low Deviation Factor



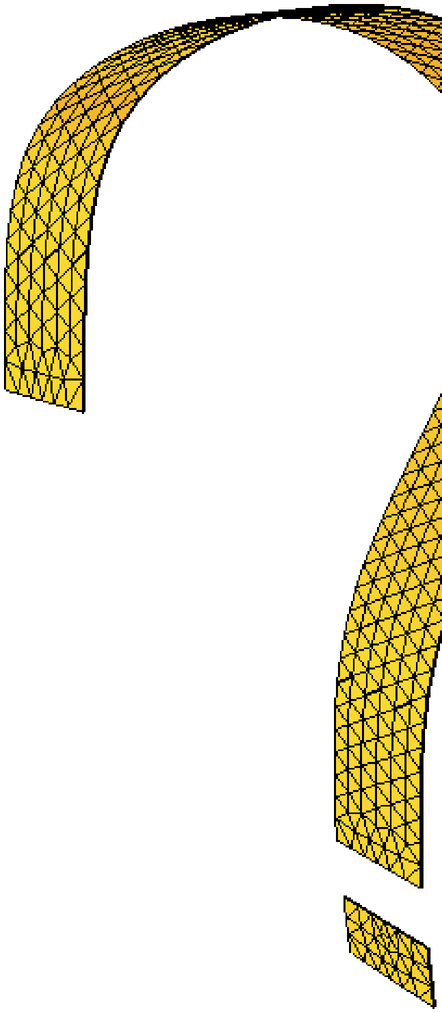


High Deviation Factor

References

- 1. Kreyszig, E. *Differential Geometry*. New York: Dover, P.34, 1991.
- 2. Gray, A. “Osculating Circles to Plane Curves.” *Modern Differential Geometry of Curves and Surfaces with Mathematica, 2nd ed.* Boca Raton, FL: CRC Press, pp. 111-115, 1997.
- 3. Durell, C.V. *Modern Geometry: The Straight Line and Circle*, London: Macmillan, pp. 19-20, 1928.
- 4. Piegl, L. and Tiller, W. *The NURBS Book, 2nd ed.* Heidelberg: Springer-Verlag, 1997.
- 5. Farin, G. *Curves and Surfaces for CAGD: A Practical Guide, 5th ed.*, London: Academic Press, 2002.
- 6. Gray, A. *Modern Differential Geometry of Curves and Surfaces*, Boca Raton, FL: CRC Press, 1993.

Questions?



www.hpc.msstate.edu

Author Information

- David O. McLaurin, Ph.D
- Title: Assistant Research Professor
- Company: Mississippi State University
- Phone: (662) 325-7296
- Email: d.mclaurin@msstate.edu
- 200 Research Blvd Starkville, MS 39759