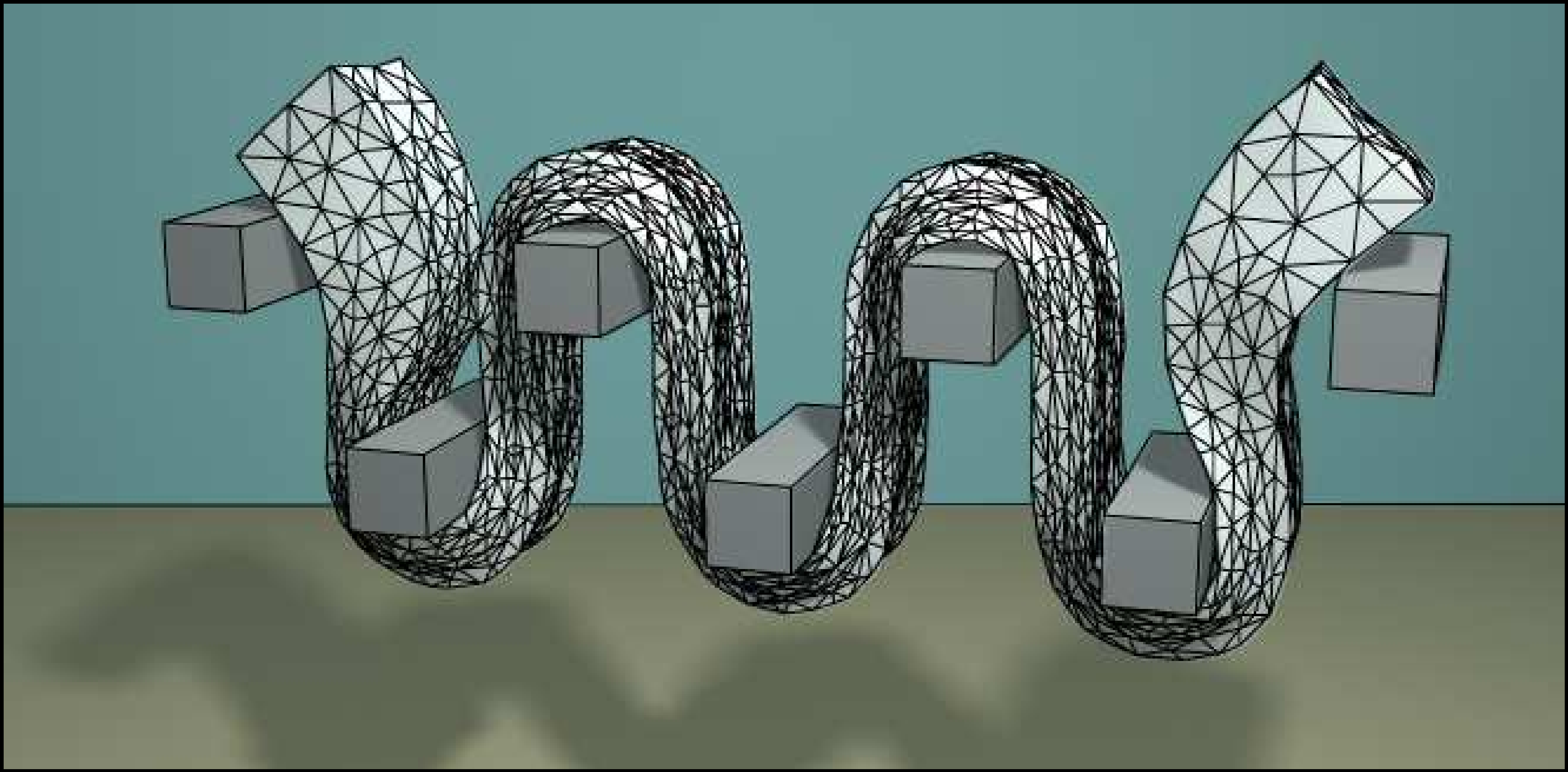
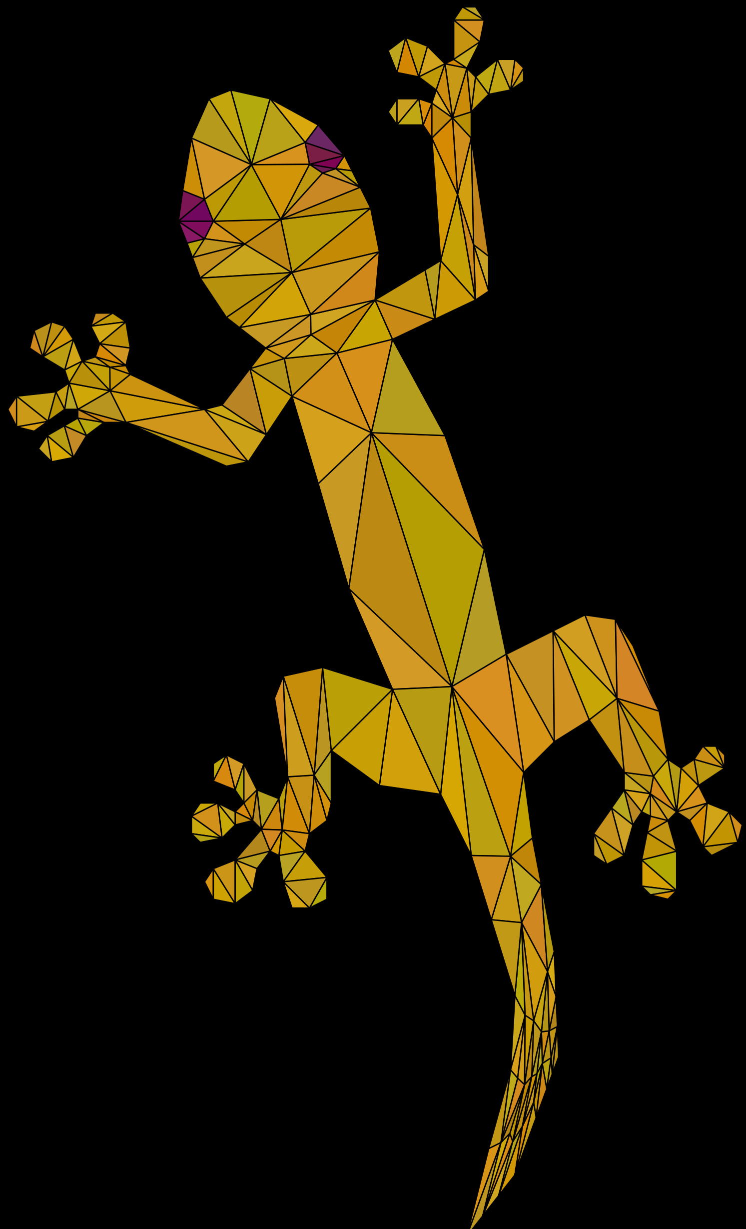
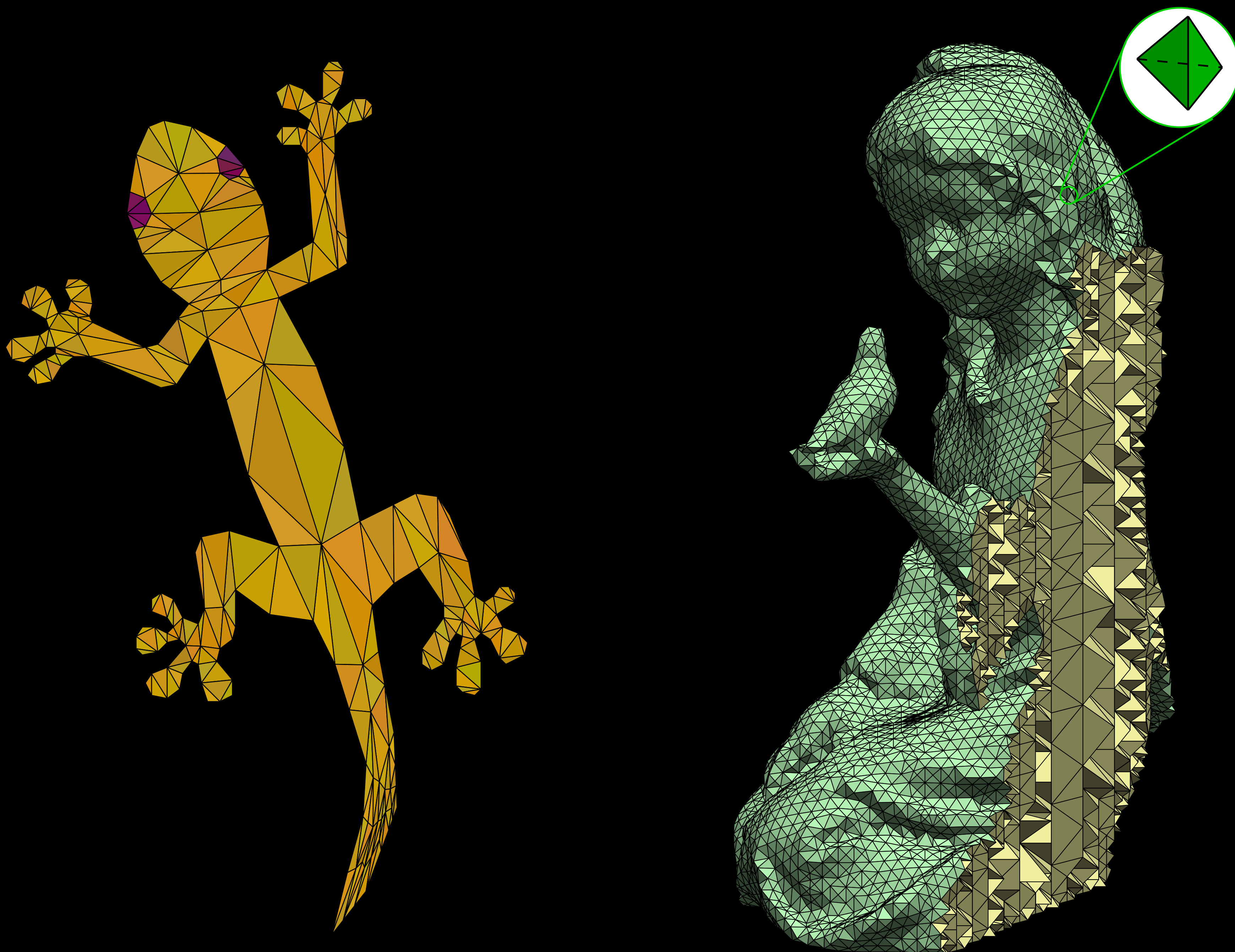
Dynamic Local Remeshing for Elastoplastic Simulation

(and for Just Plain Really Great Tetrahedral Meshes)

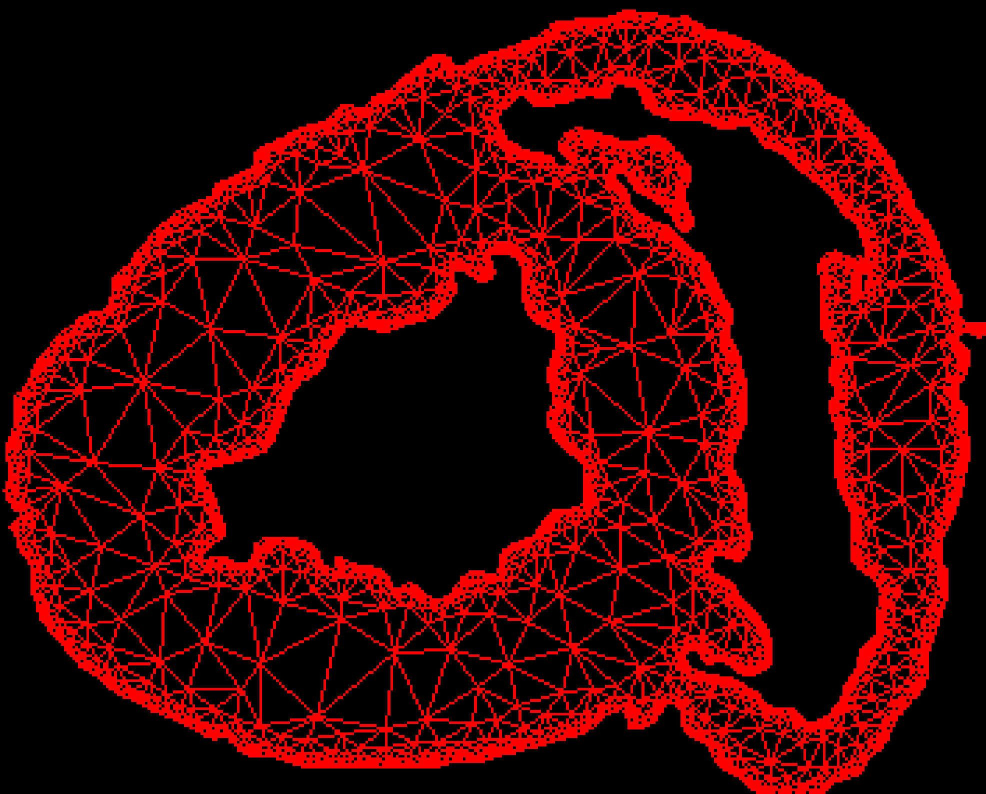
Jonathan Shewchuk, University of California, Berkeley





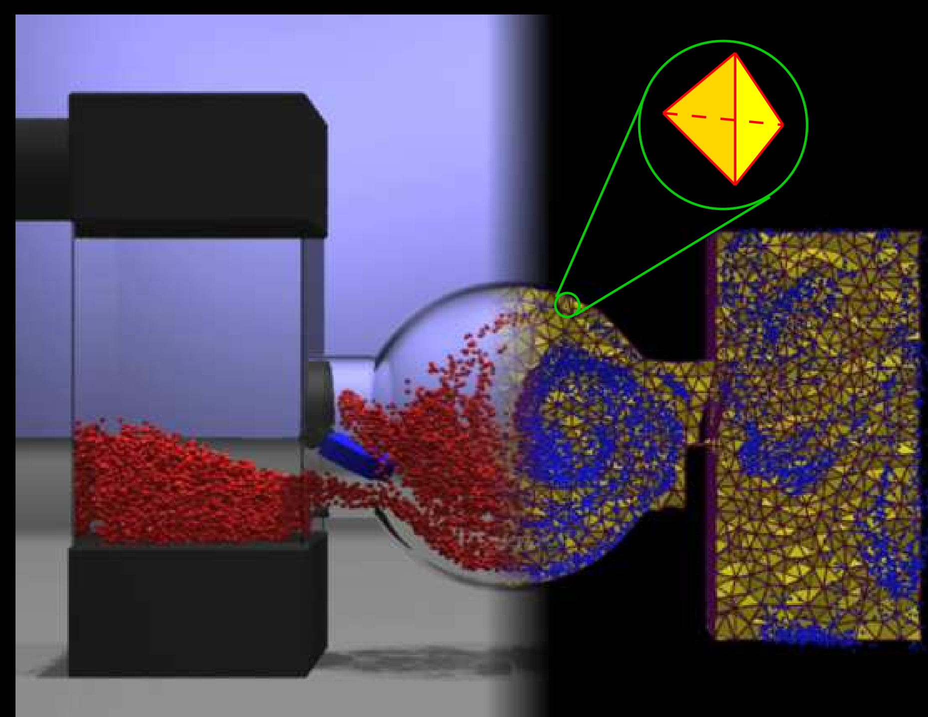


Finite Element Methods for Solving Partial Differential Equations

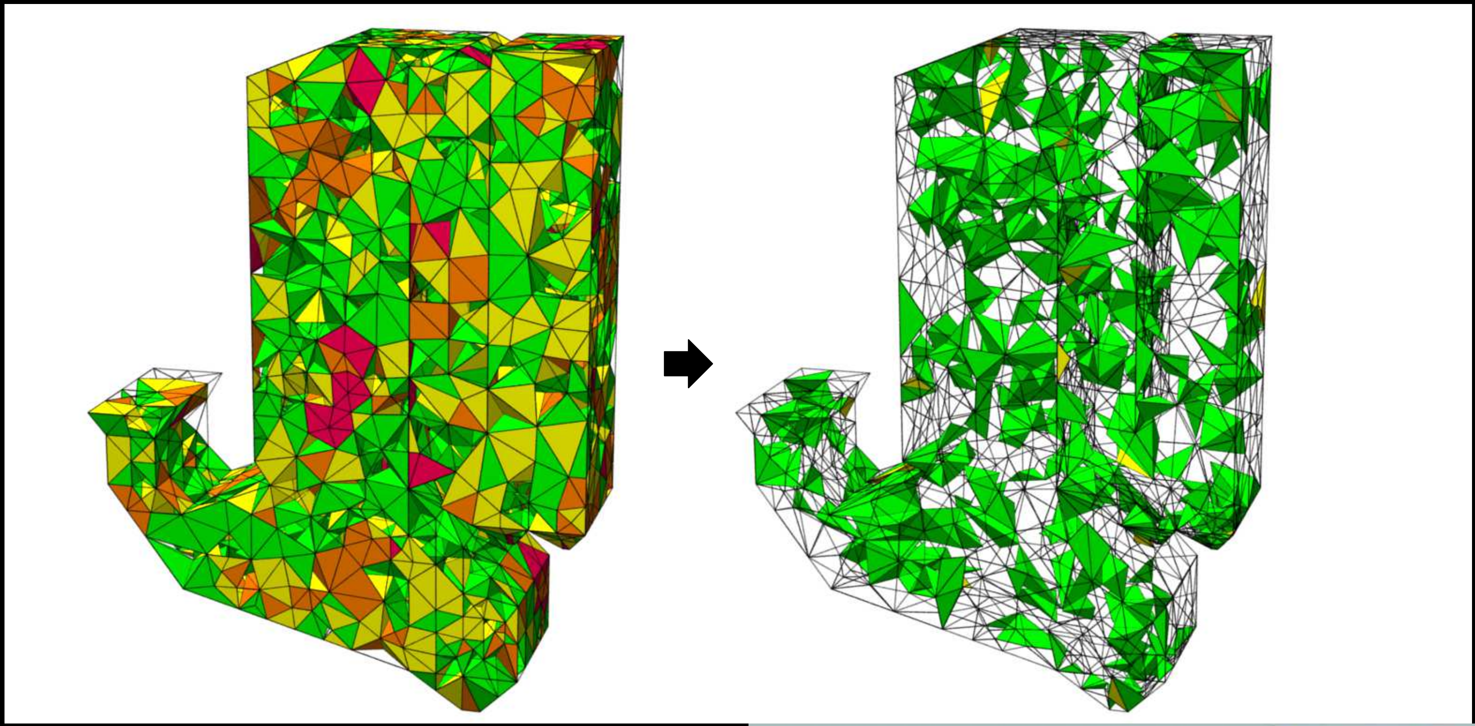


Used to simulate mechanical deformation, fluid flow, heat transfer, electrical propagation, other physical phenomena.

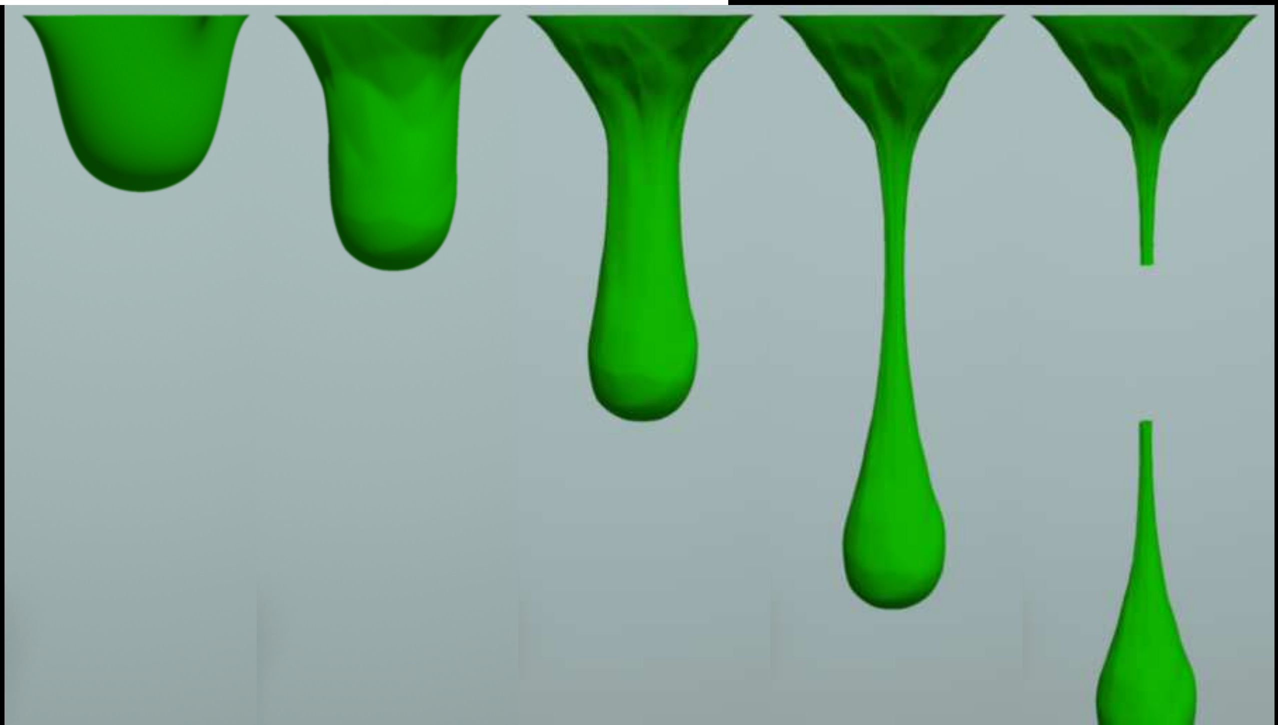
Simulated object is partitioned into a *mesh* of *elements;* frequently triangles or tetrahedra.



Today’s Topics



1.
Mesh
Improvement

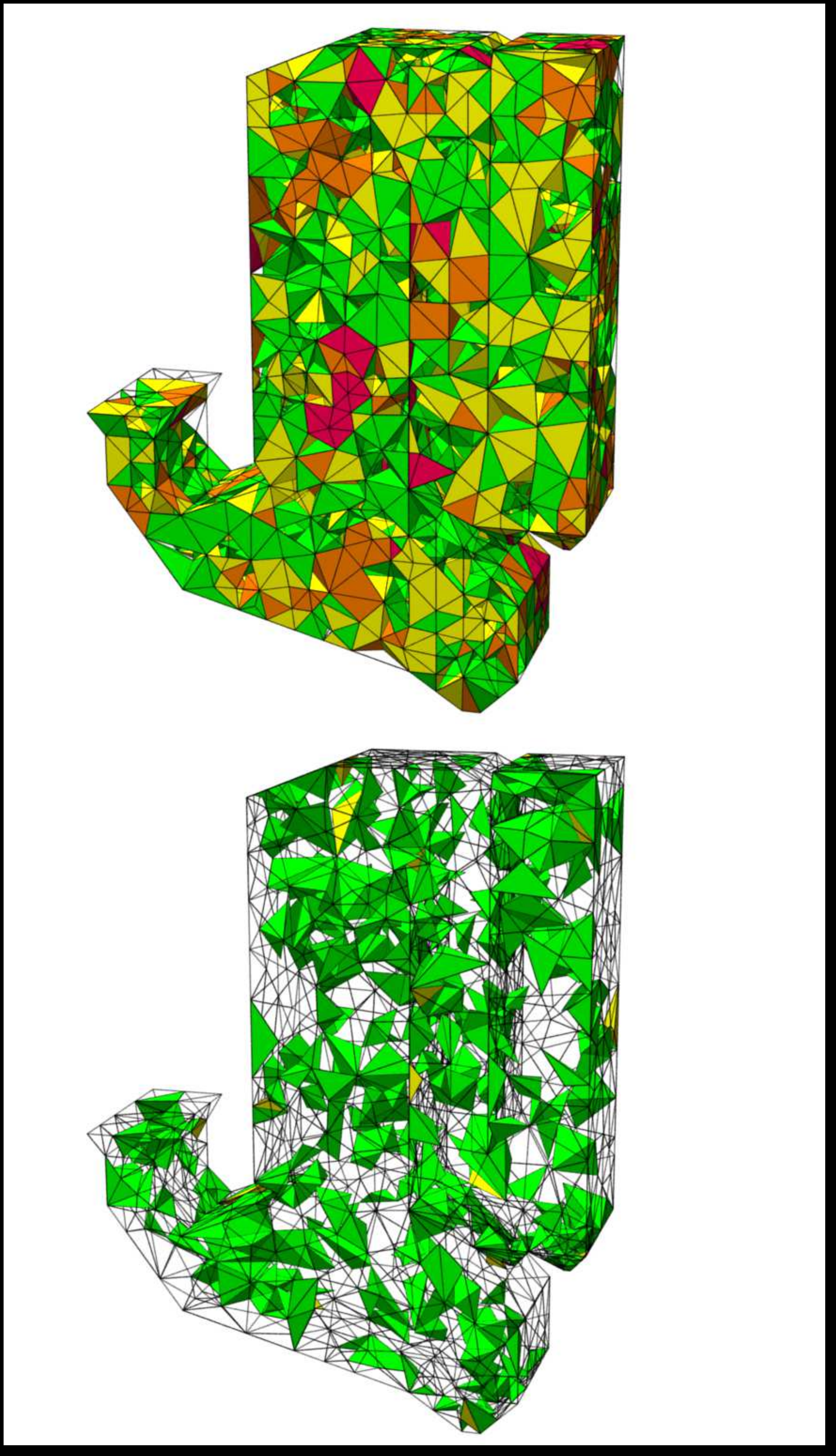


2.
Plastic Flow
Simulation

Aggressive Tetrahedral Mesh Improvement

Bryan Klingner
Jonathan Shewchuk

Computer Science Division
University of California
Berkeley, California



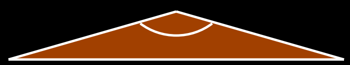
Bryan Klingner

How Angles Affect Applications

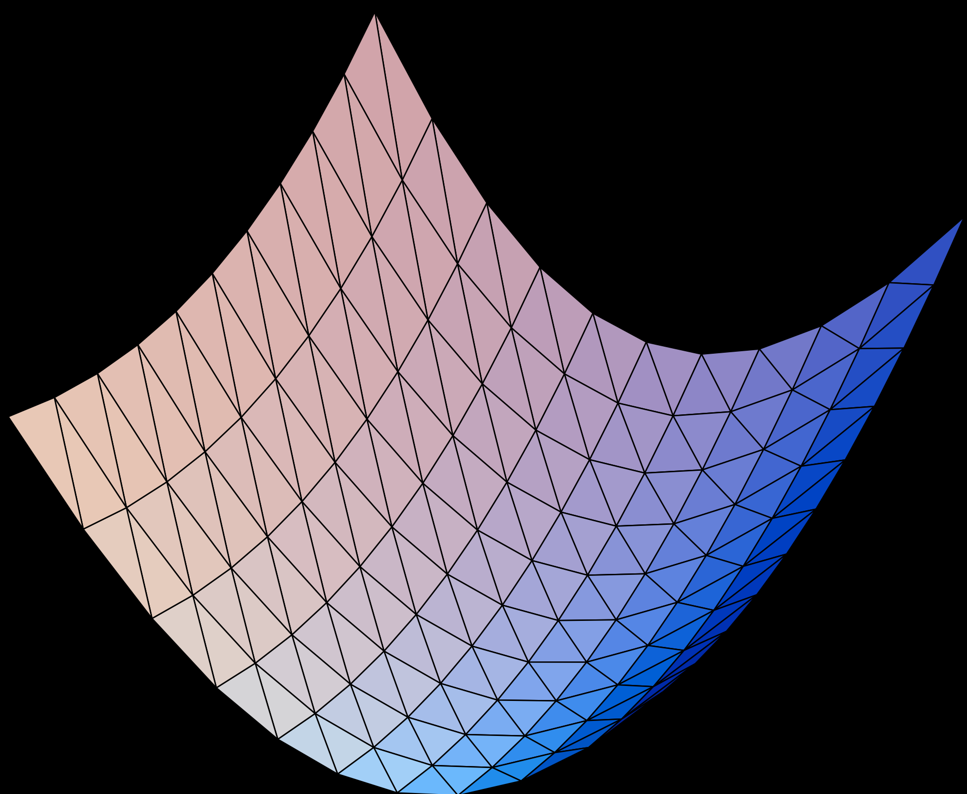
Skinny elements cause problems.

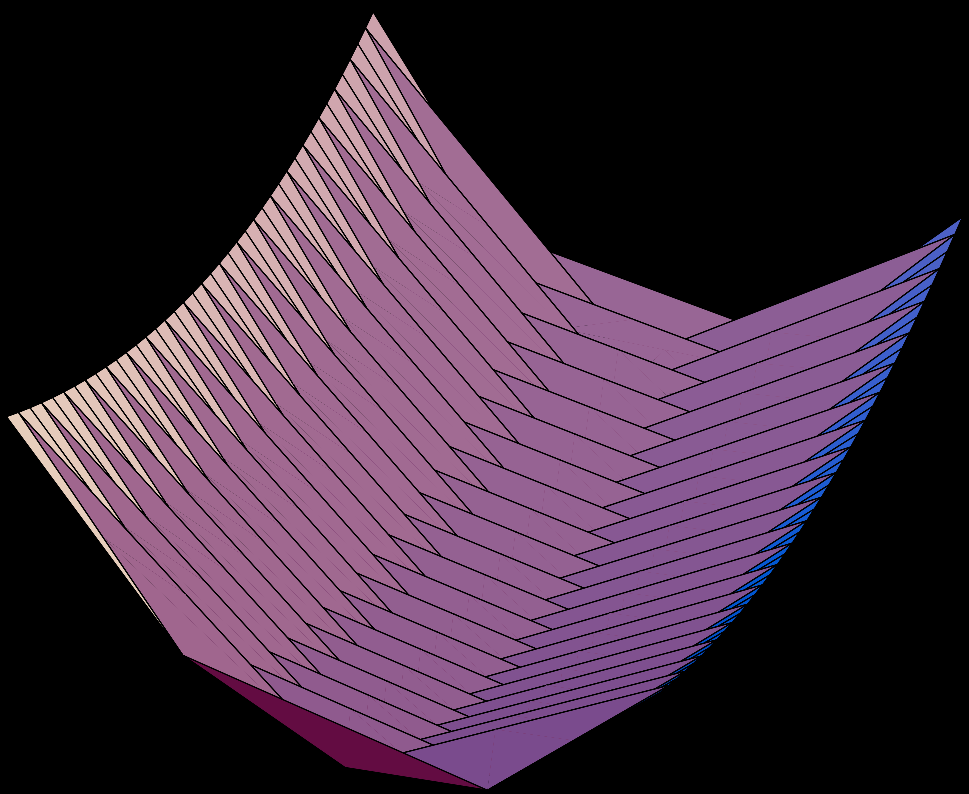
How Angles Affect Applications

Skinny elements cause problems.



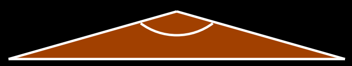
Large angles cause discretization errors & big errors in interpolated derivatives.



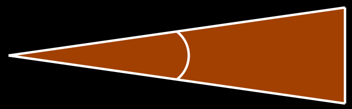


How Angles Affect Applications

Skinny elements cause problems.



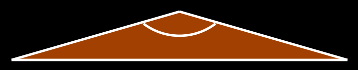
Large angles cause discretization errors & big errors in interpolated derivatives.



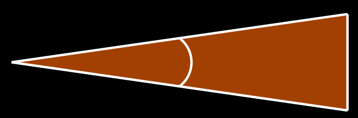
Small angles cause poor conditioning.

How Angles Affect Applications

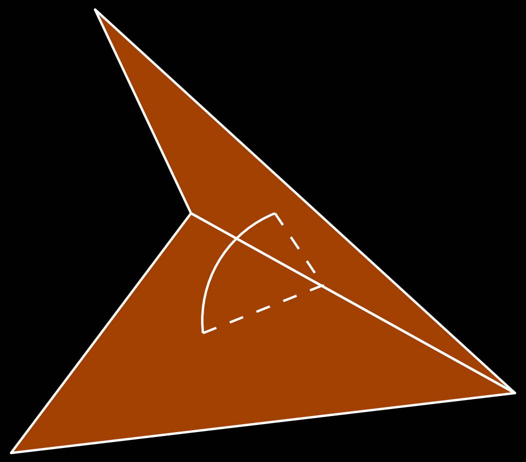
Skinny elements cause problems.



Large angles cause discretization errors & big errors in interpolated derivatives.



Small angles cause poor conditioning.

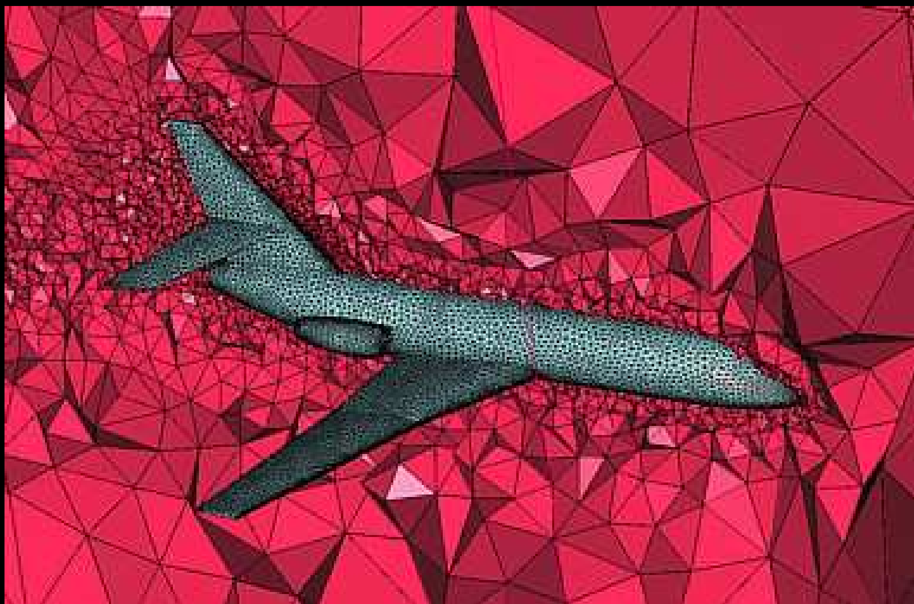


For tetrahedra, this applies to the dihedral angles.

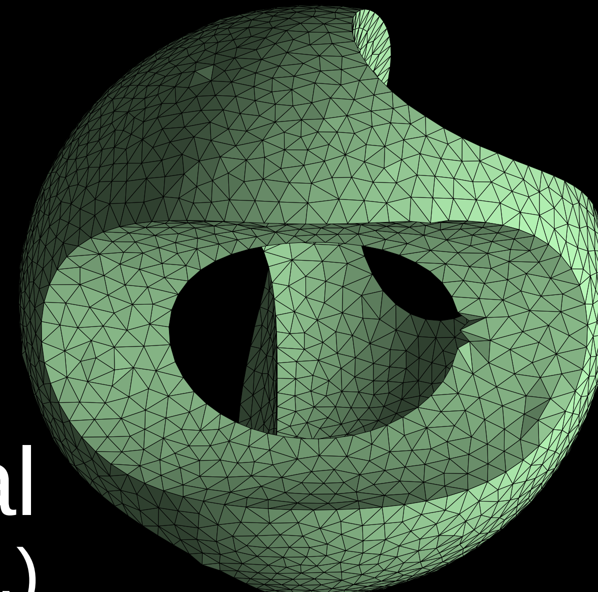
(Not the plane angles!)

Previous Work

Heuristic

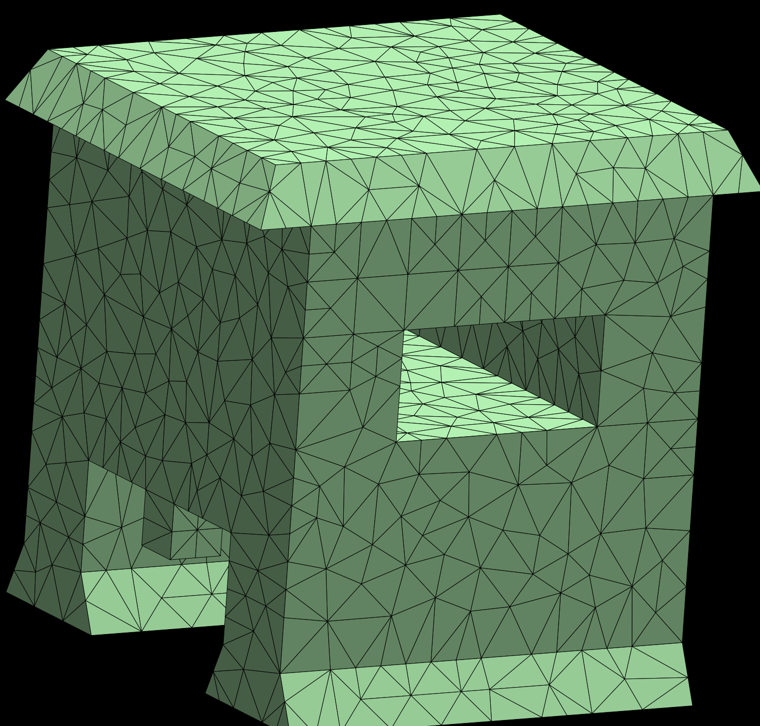


Delaunay
(George et al.)



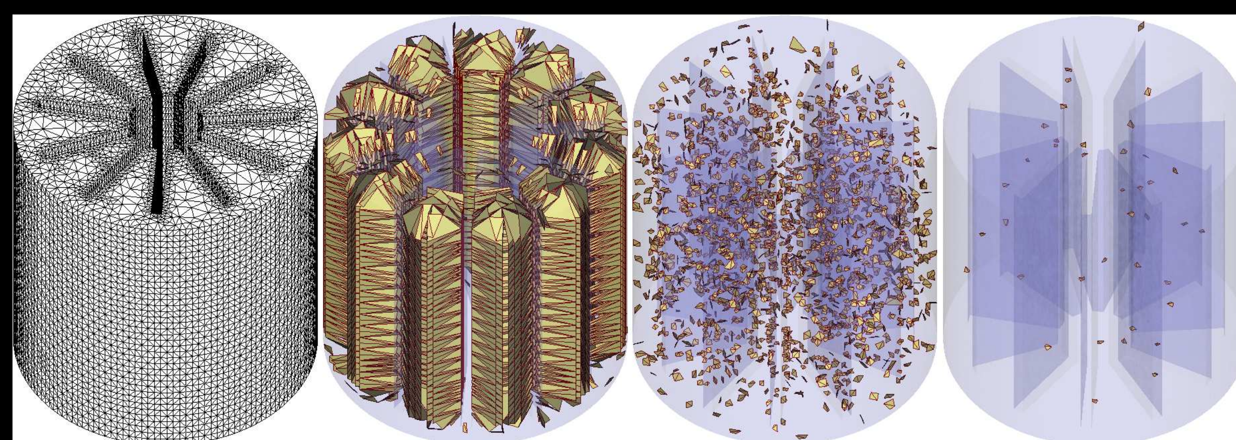
Variational
(Alliez et al.)

Theoretical



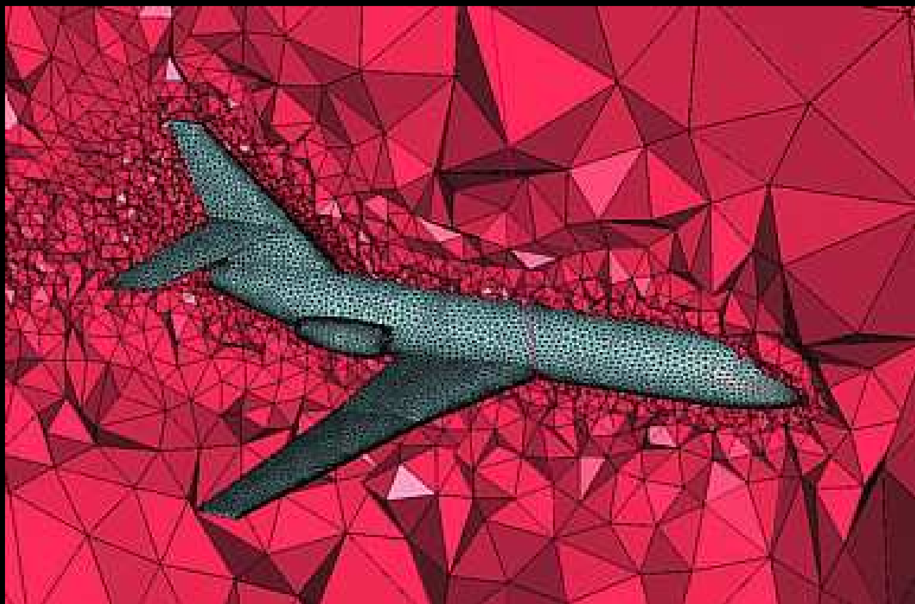
Delaunay
refinement (me)

Sliver exudation (Cheng et al.)



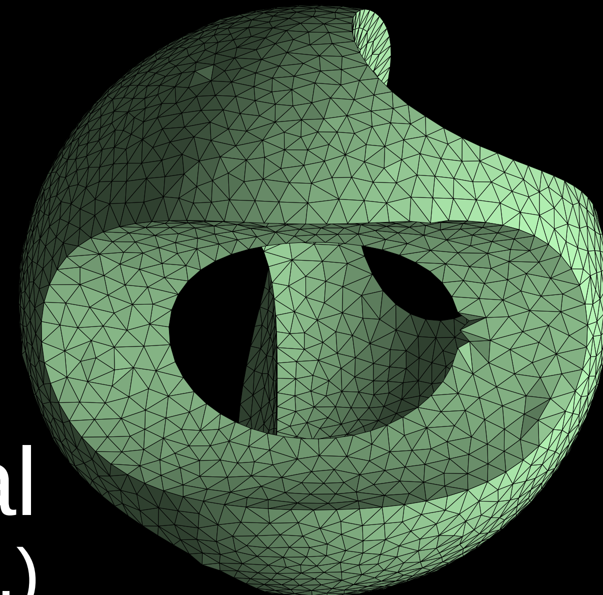
Previous Work

Heuristic



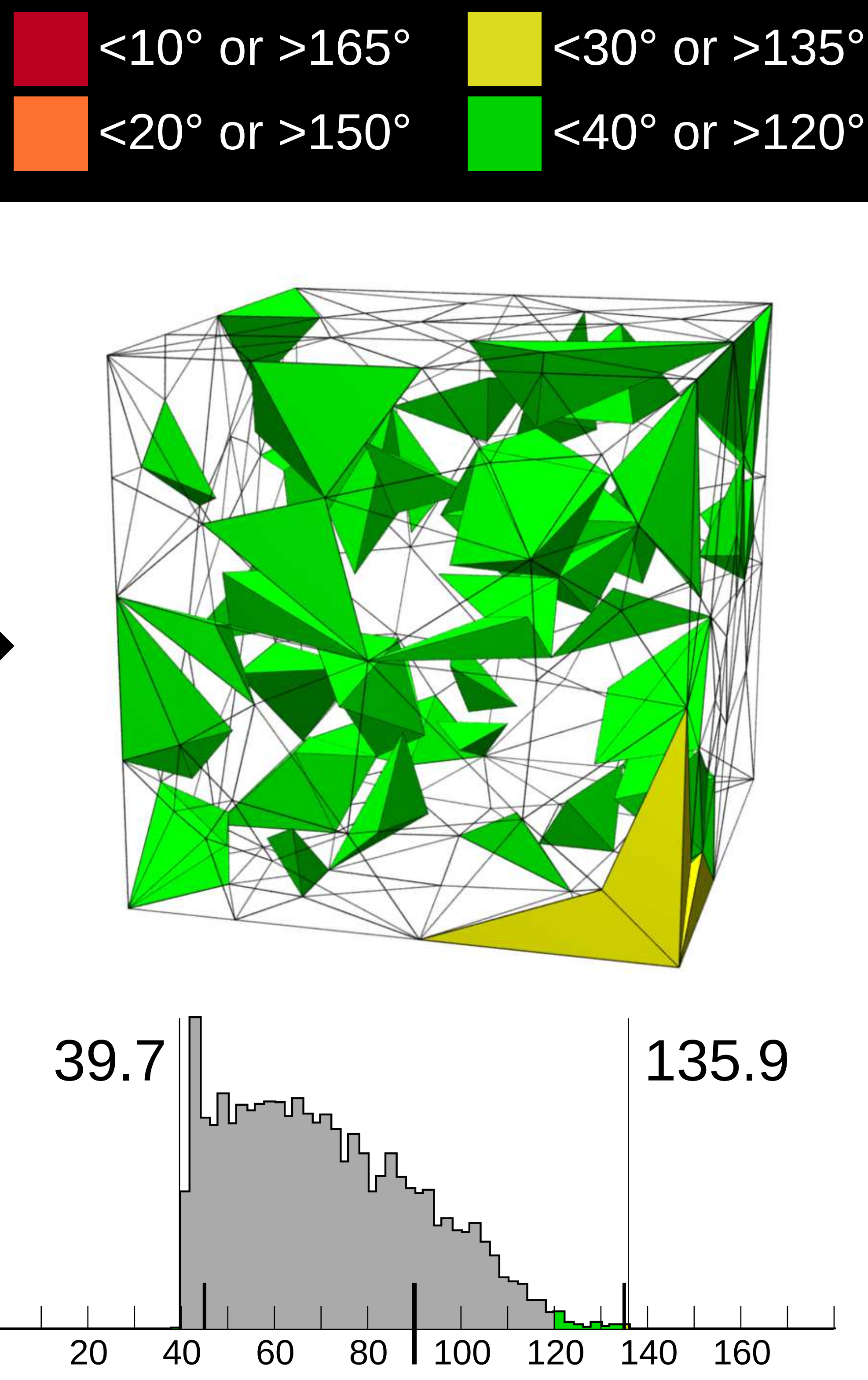
Delaunay
(George et al.)

Variational
(Alliez et al.)



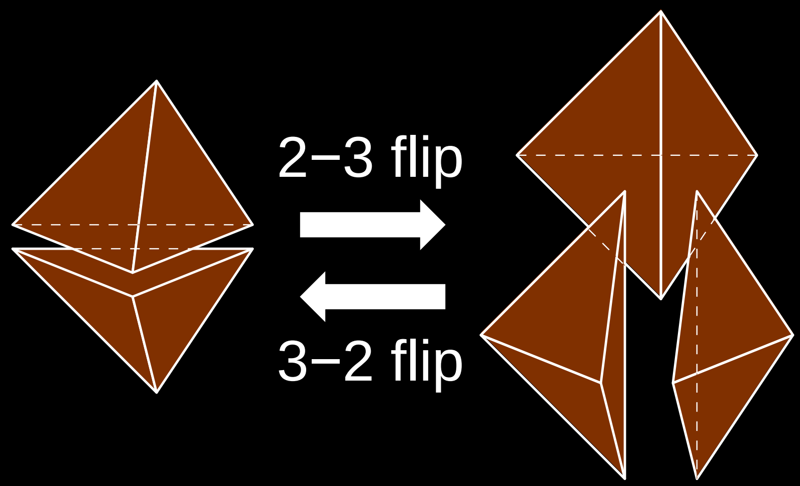
One bad tetrahedron can ruin a simulation!

Goal: Improve

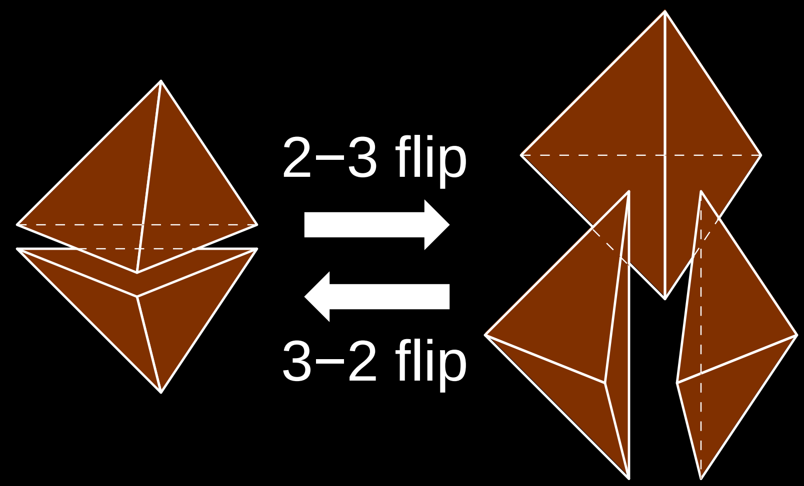


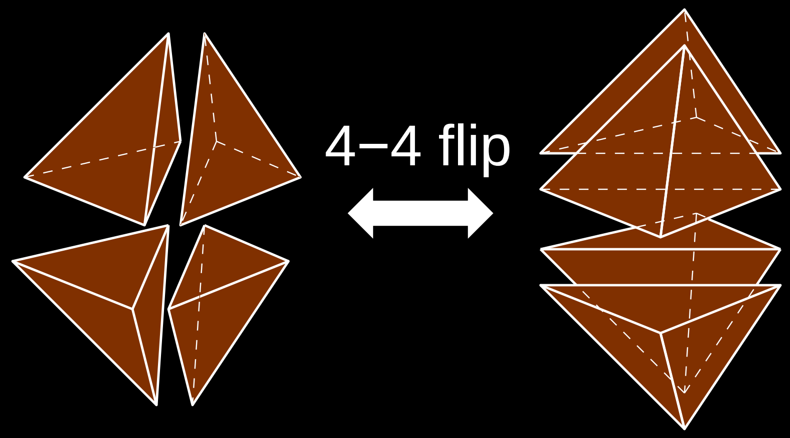
Methods

Topological Transformations

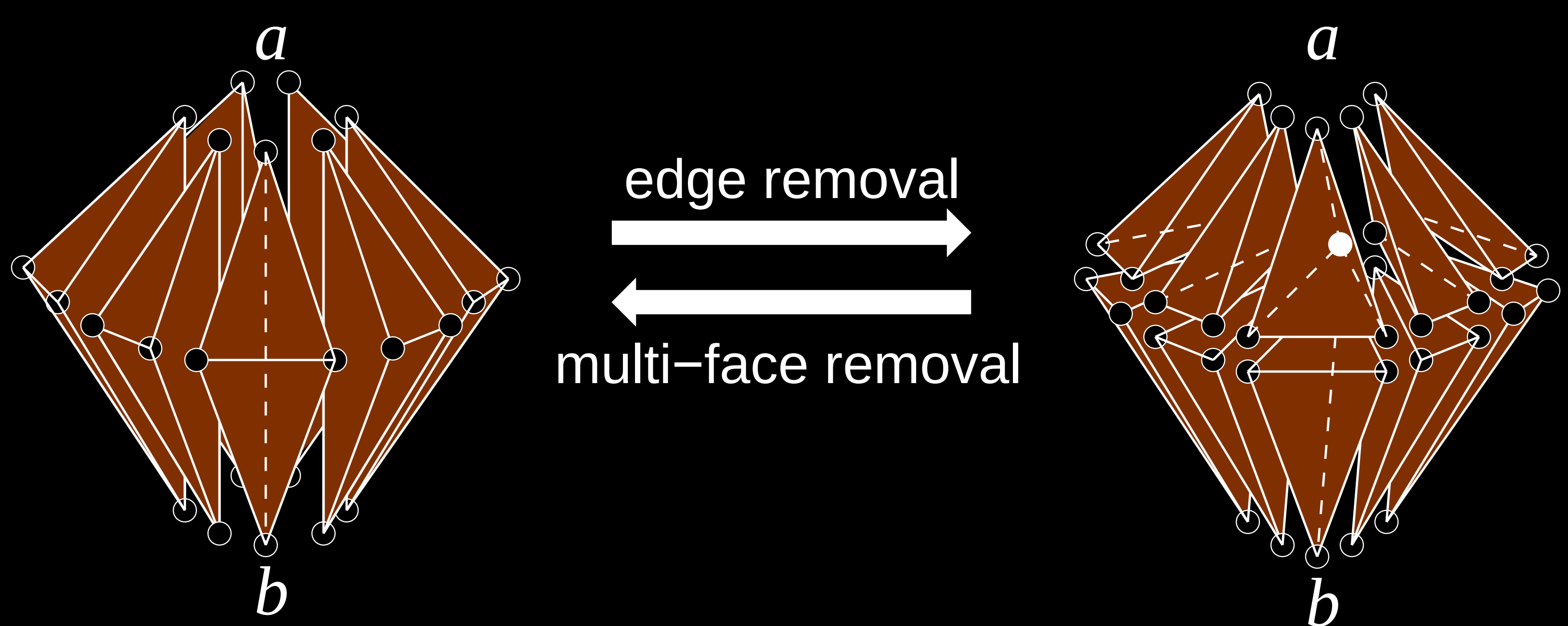


Topological Transformations



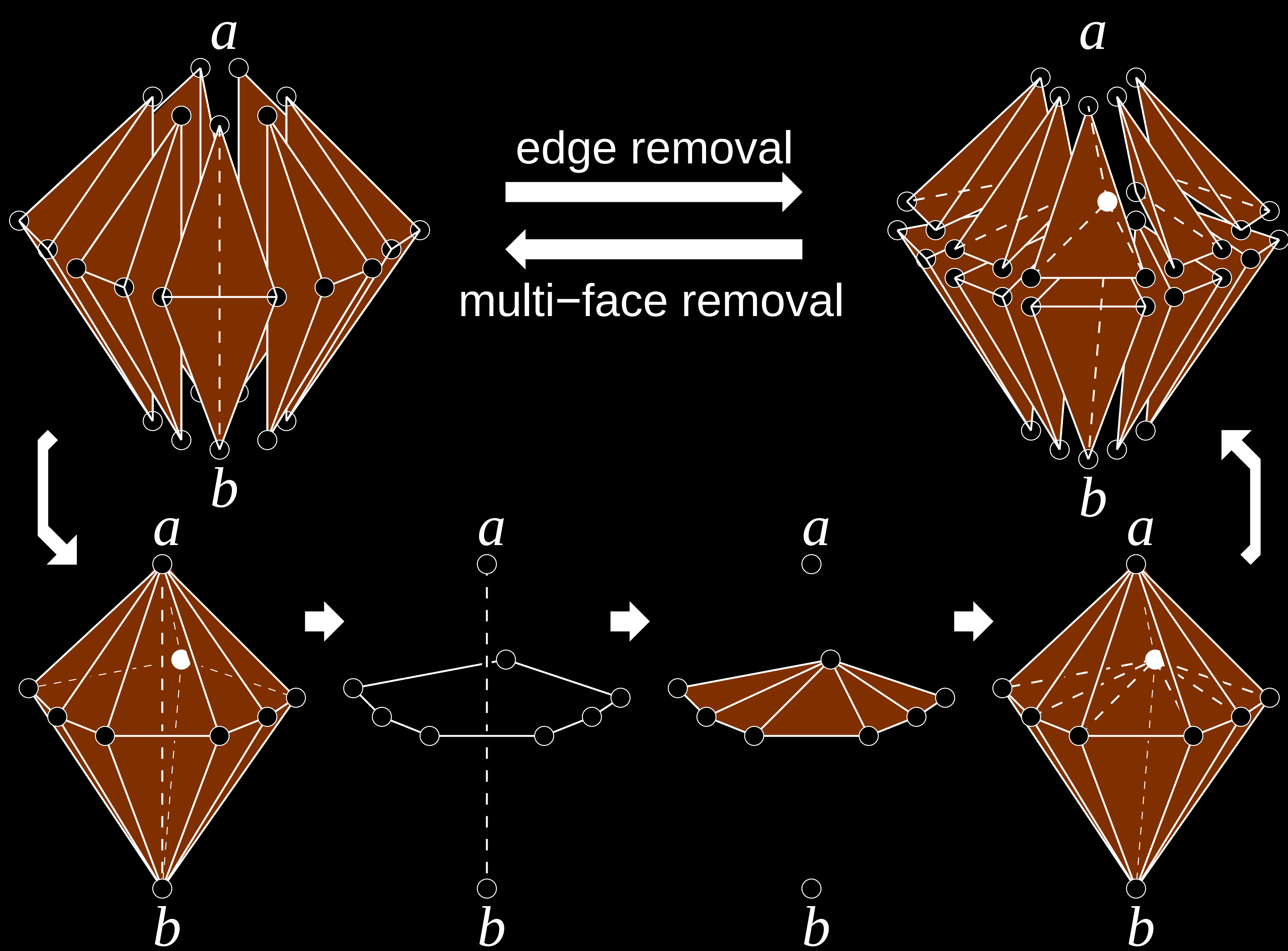


Topological Transformations



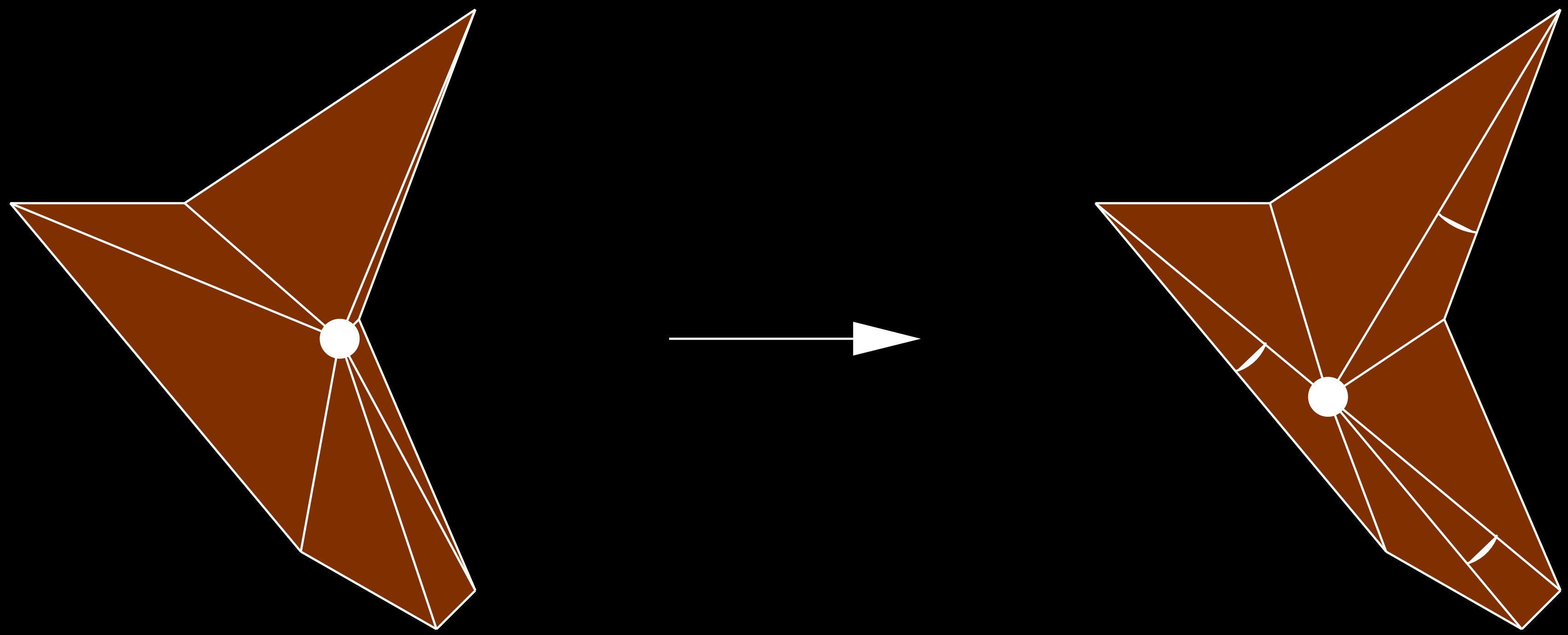
(Combinatorial optimization.)

Topological Transformations



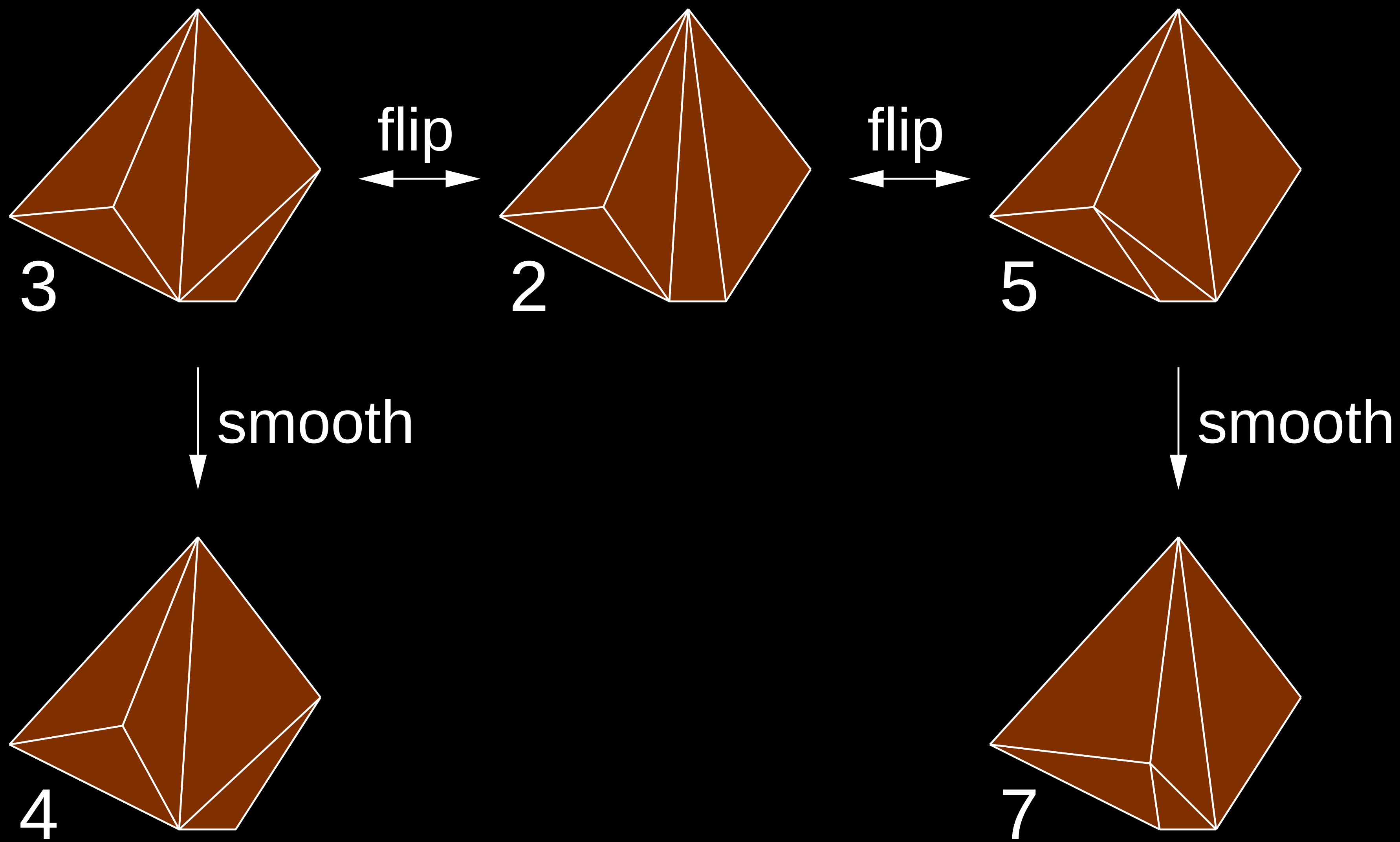
(Combinatorial optimization.)

Optimization–Based Smoothing



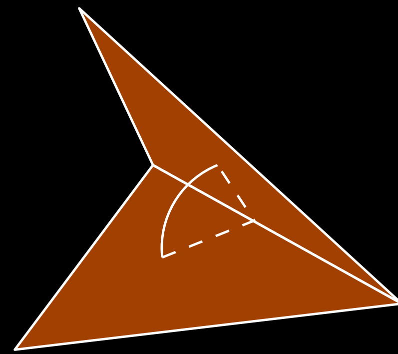
(Numerical optimization.)

Hill Climbing Optimization

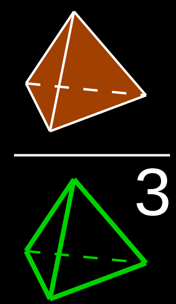


Tetrahedron Quality Measures

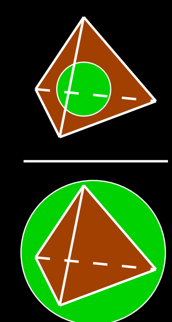
1. Minimum sine of the six dihedral angles.



2. $$\frac{\text{Volume}}{(\text{Root−mean−squared edge length})^3}$$



3. $$\frac{\text{Radius of inscribed sphere}}{\text{Radius of circumscribing sphere}}$$



Mesh Quality

Quality vector of a mesh:
sorted list of its tetrahedra’s quality scores.

<0.01, 0.03, 0.04, 0.08, 0.10, 0.18, 0.22, ...>

Mesh Quality

Quality vector of a mesh:
sorted list of its tetrahedra's quality scores.

<0.01, 0.03, 0.04, 0.08, 0.10, 0.18, 0.22, ...>

Another mesh is better if its quality vector is *lexicographically greater.*

<0.01, 0.03, 0.08, 0.10, 0.15, 0.17, 0.18, ...>

Mesh Quality

Quality vector of a mesh:
sorted list of its tetrahedra's quality scores.

<0.01, 0.03, 0.04, 0.08, 0.10, 0.18, 0.22, ...>

Another mesh is better if its quality vector is *lexicographically greater.*

<0.01, 0.03, 0.08, 0.10, 0.15, 0.17, 0.18, ...>

Only perform operations that improve the quality vector. → No cycles.

Old Ideas and New

Our starting point: Lori Freitag and Carl Ollivier−Gooch, "Tetrahedral Mesh Improvement Using Swapping and Smoothing," 1997.

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING, VOL. 40, 3979–4002 (1997)

TETRAHEDRAL MESH IMPROVEMENT USING SWAPPING AND SMOOTHING

LORI A. FREITAG[1,*] AND CARL OLLIVIER-GOOCH[2]

[1] *Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Illinois 60439, U.S.A.*
[2] *Department of Mechanical Engineering, University of British Columbia, Vancouver, BC V6T1Z4 Canada*

ABSTRACT

Automatic mesh generation and adaptive refinement methods for complex three-dimensional domains have proven to be very successful tools for the efficient solution of complex applications problems. These methods can, however, produce poorly shaped elements that cause the numerical solution to be less accurate and more difficult to compute. Fortunately, the shape of the elements can be improved through several mechanisms, including face- and edge-swapping techniques, which change local connectivity, and optimization-based mesh smoothing methods, which adjust mesh point location. We consider several criteria for each of these two methods and compare the quality of several meshes obtained by using different combinations of swapping and smoothing. Computational experiments show that swapping is critical to the improvement of general mesh quality and that optimization-based smoothing is highly effective in eliminating very small and very large angles. High-quality meshes are obtained in a computationally efficient manner by using optimization-based smoothing to improve only the worst elements and a smart variant of Laplacian smoothing on the remaining elements. Based on our experiments, we offer several recommendations for the improvement of tetrahedral meshes.

Int. J. Numer. Meth. Engng., 40, 3979–4002 (1997)

No. of Figures: 7. No. of Tables: 18. No. of References: 23.

KEY WORDS: mesh improvement; local reconnection; mesh smoothing; optimal smoothing

1. INTRODUCTION

The use of unstructured finite element and finite volume solution methods is increasingly common

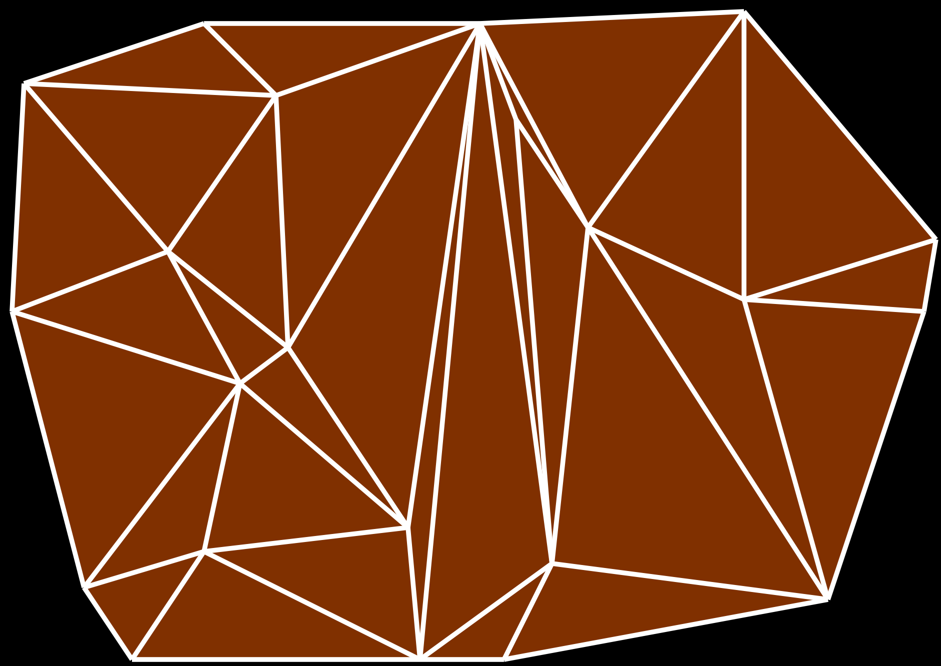
Old Ideas and New

Our starting point: Lori Freitag and Carl Ollivier−Gooch, “Tetrahedral Mesh Improvement Using Swapping and Smoothing,” 1997.

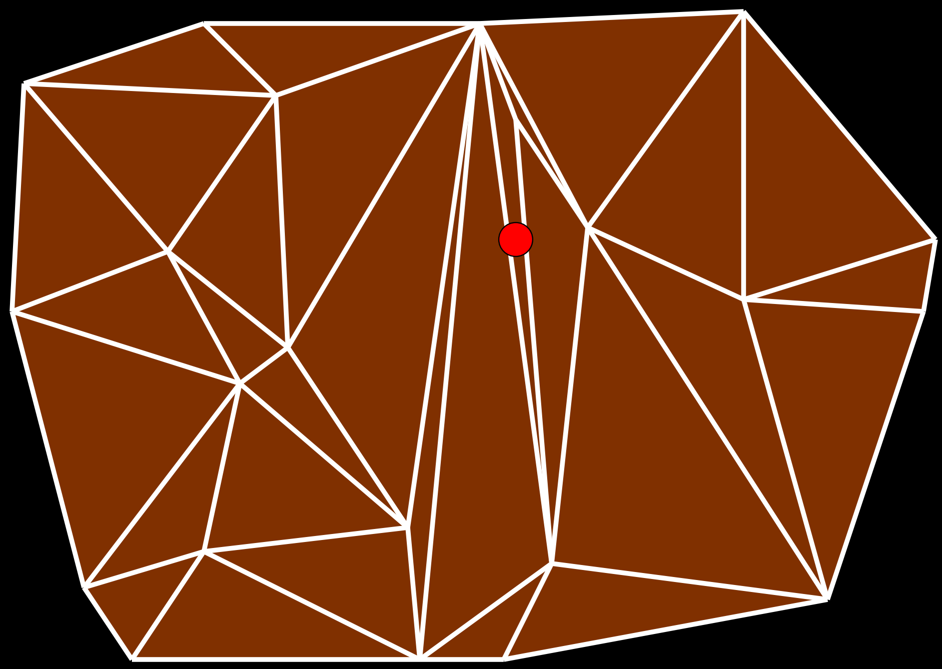
We add:

- Vertex insertion.
- Smoothing on the boundary.
- Edge removal on the boundary.
- Multi−face removal.
- Compound operation: insertion, followed by topological flips and smoothing.

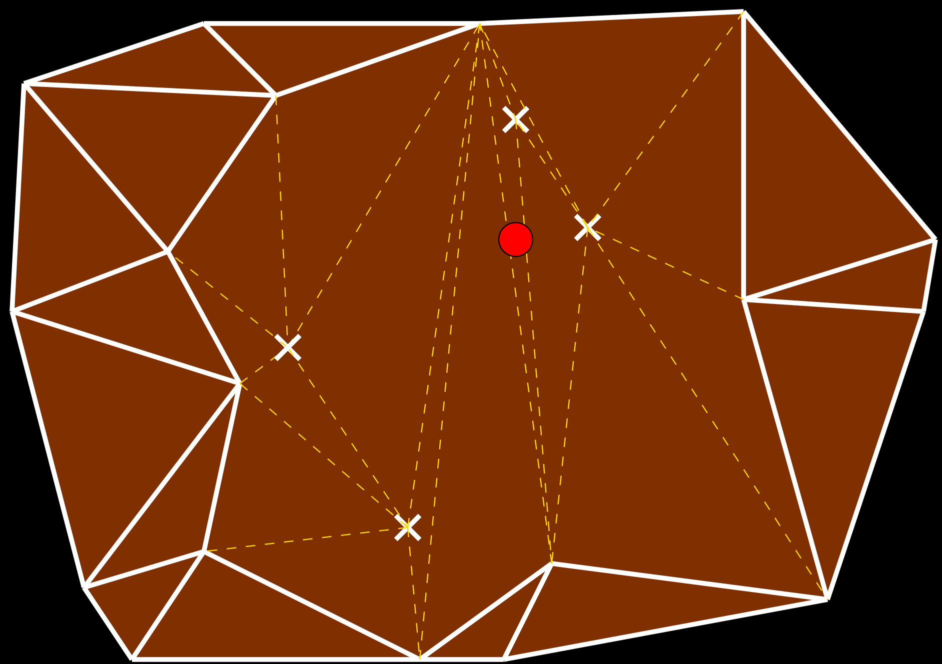
Vertex Insertion



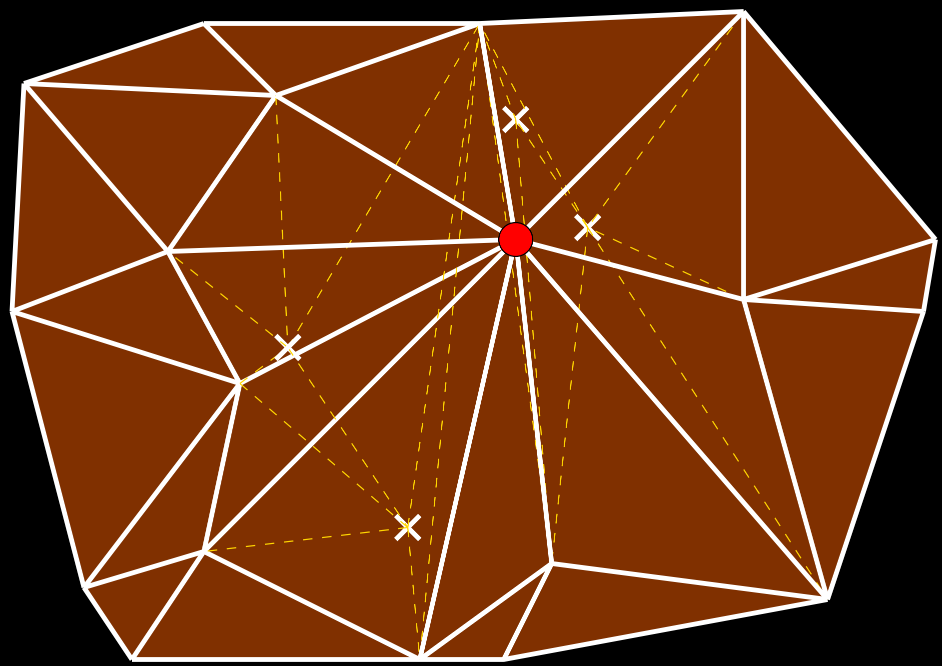
Vertex Insertion



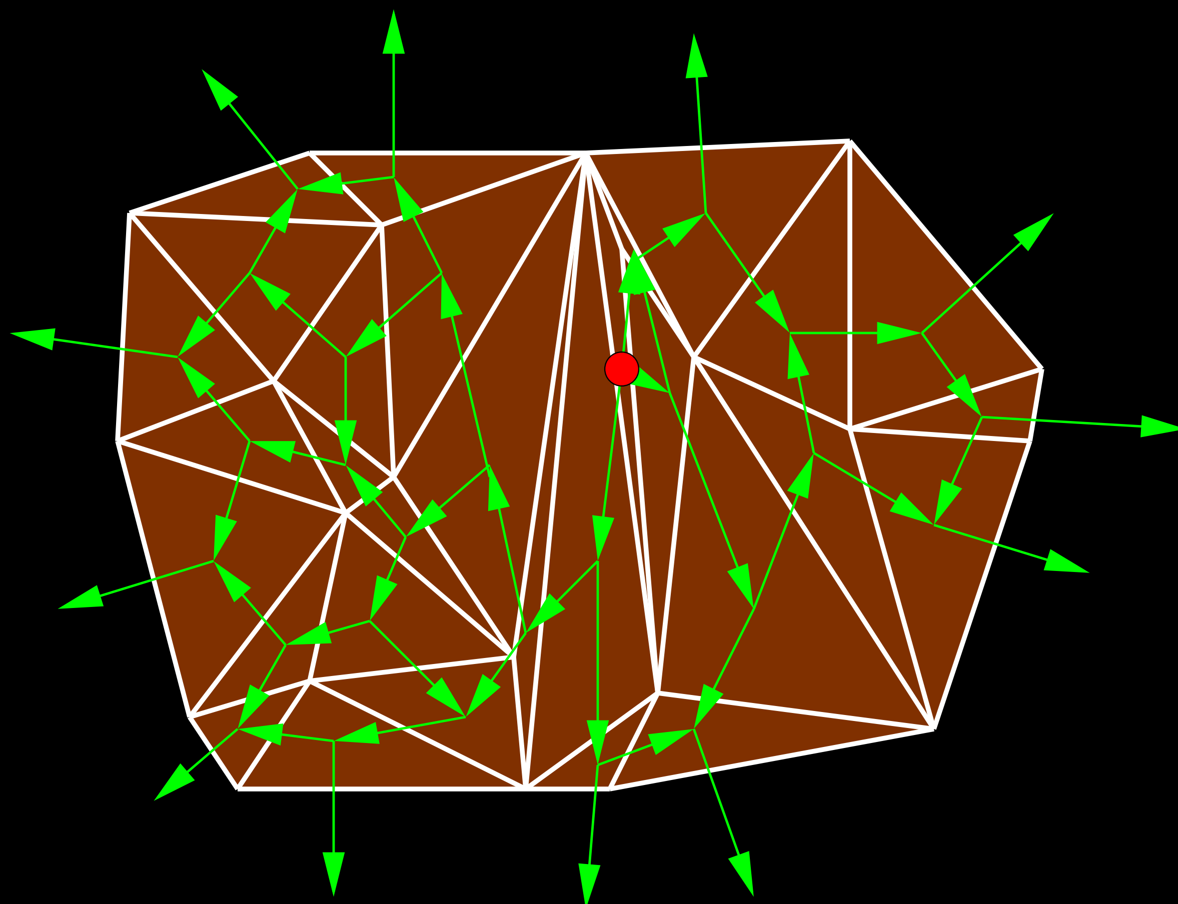
Vertex Insertion



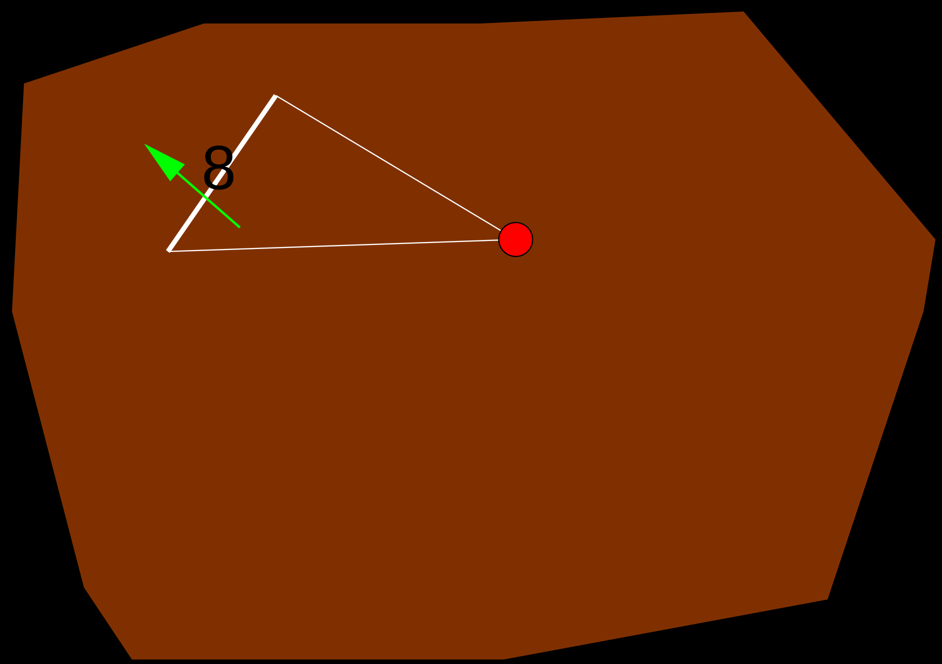
Vertex Insertion



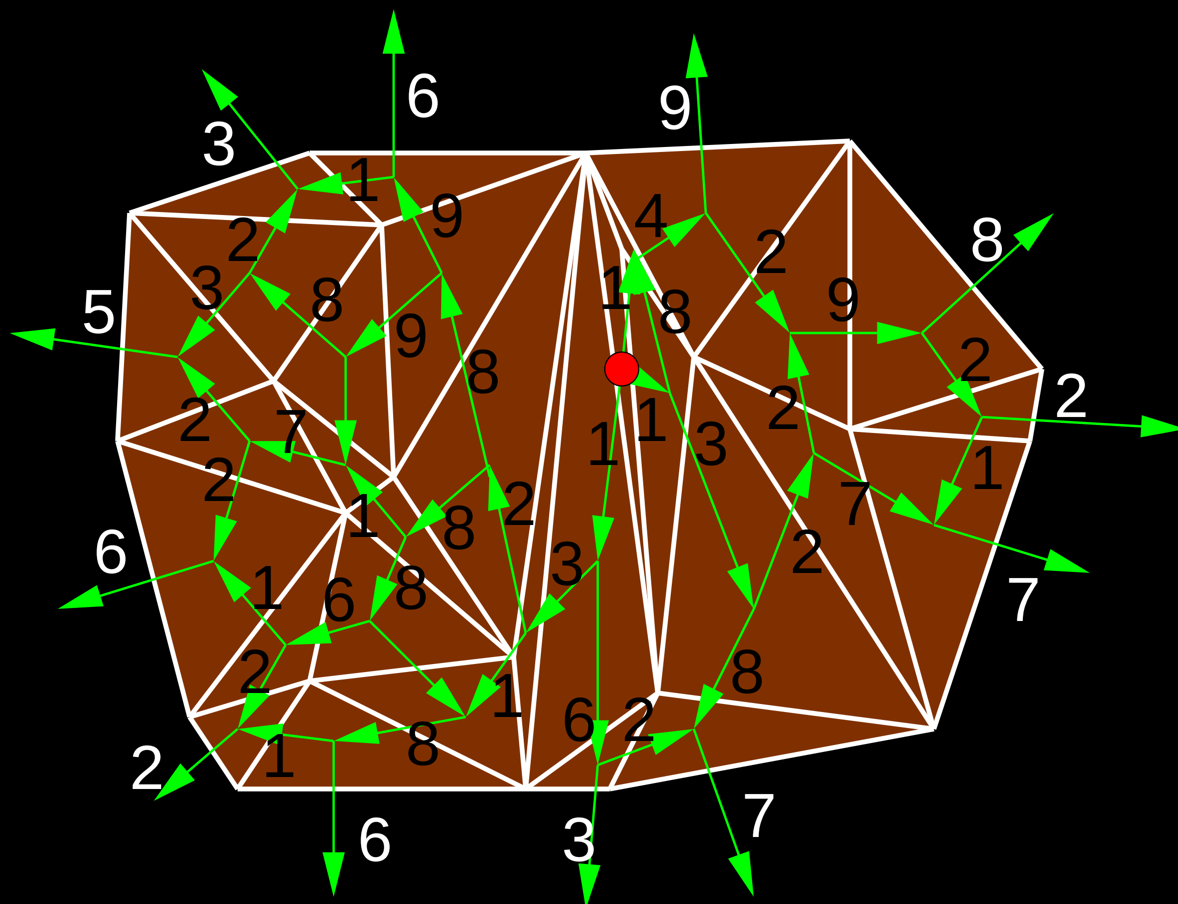
Vertex Insertion



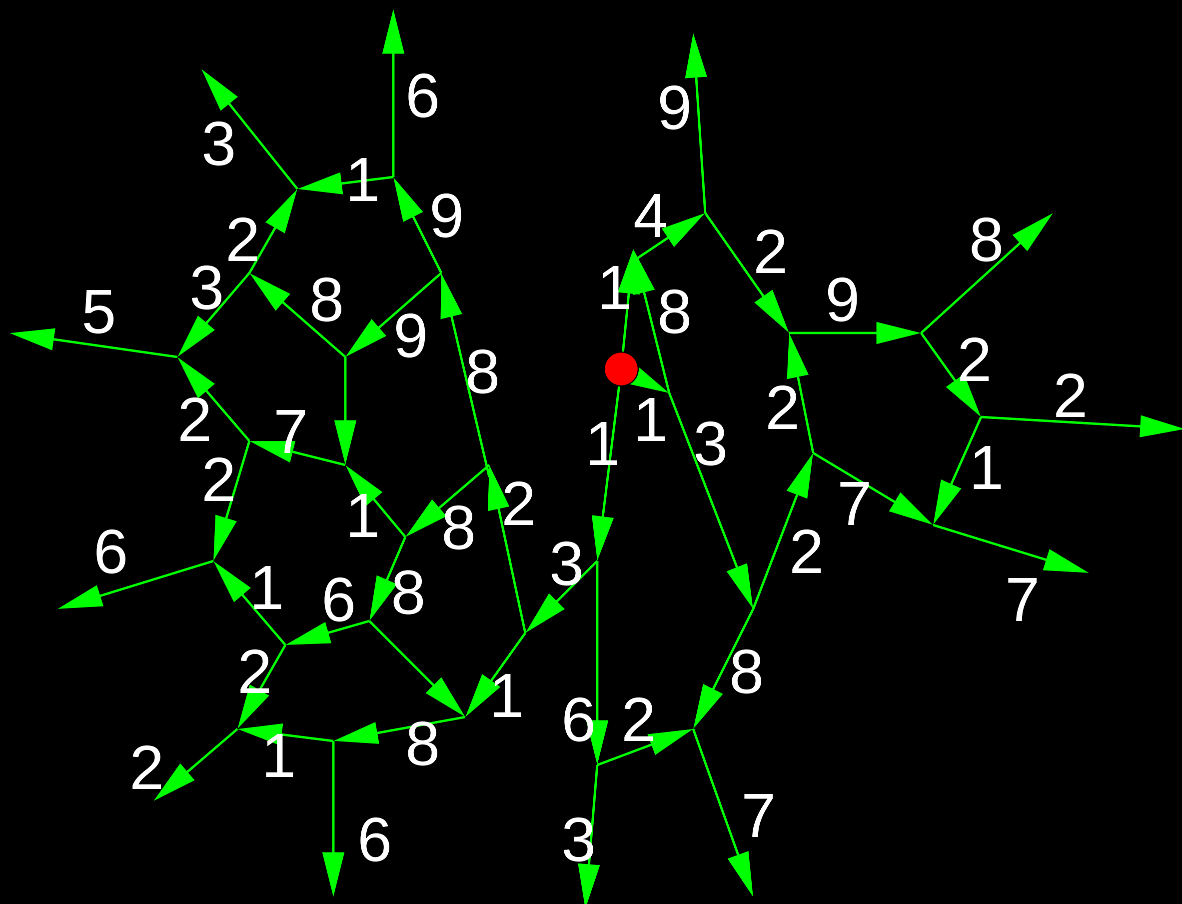
Vertex Insertion



Vertex Insertion

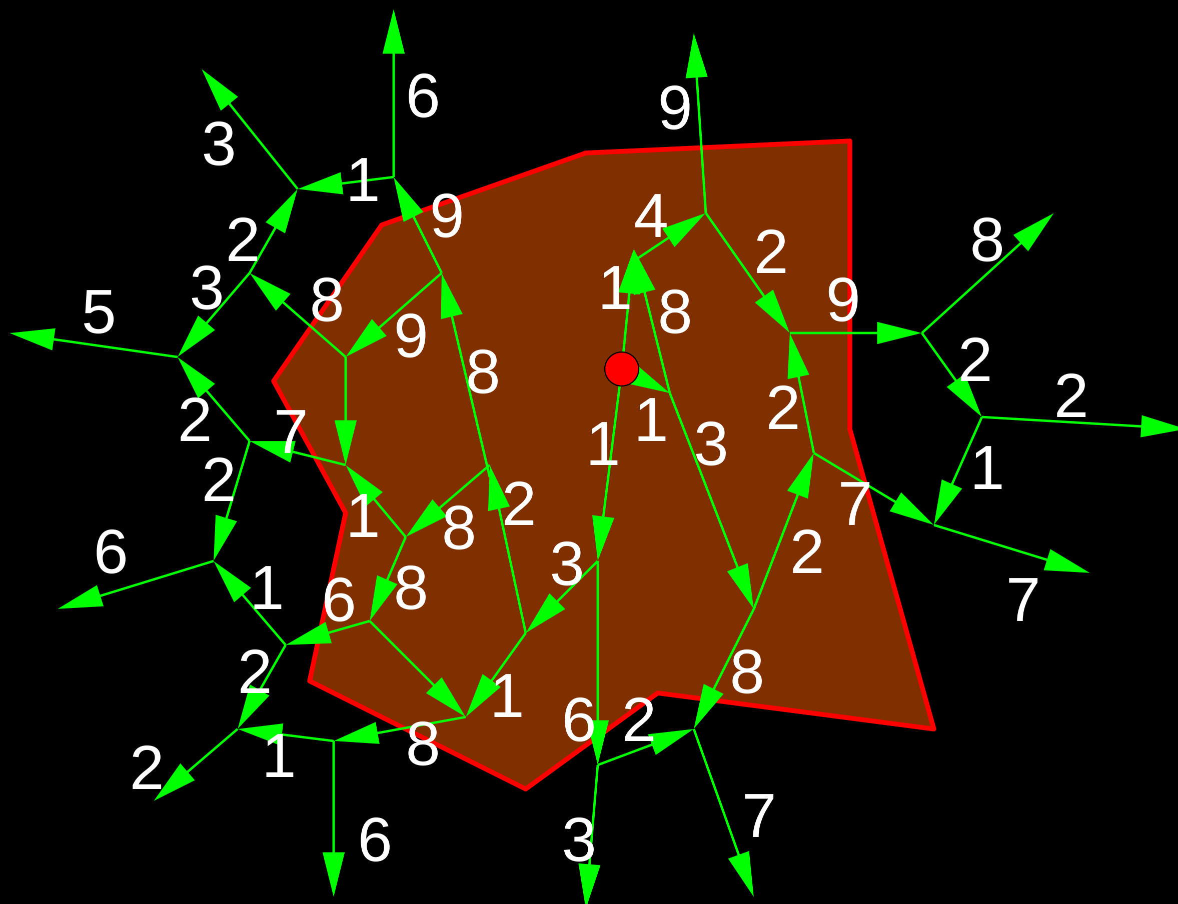


Vertex Insertion



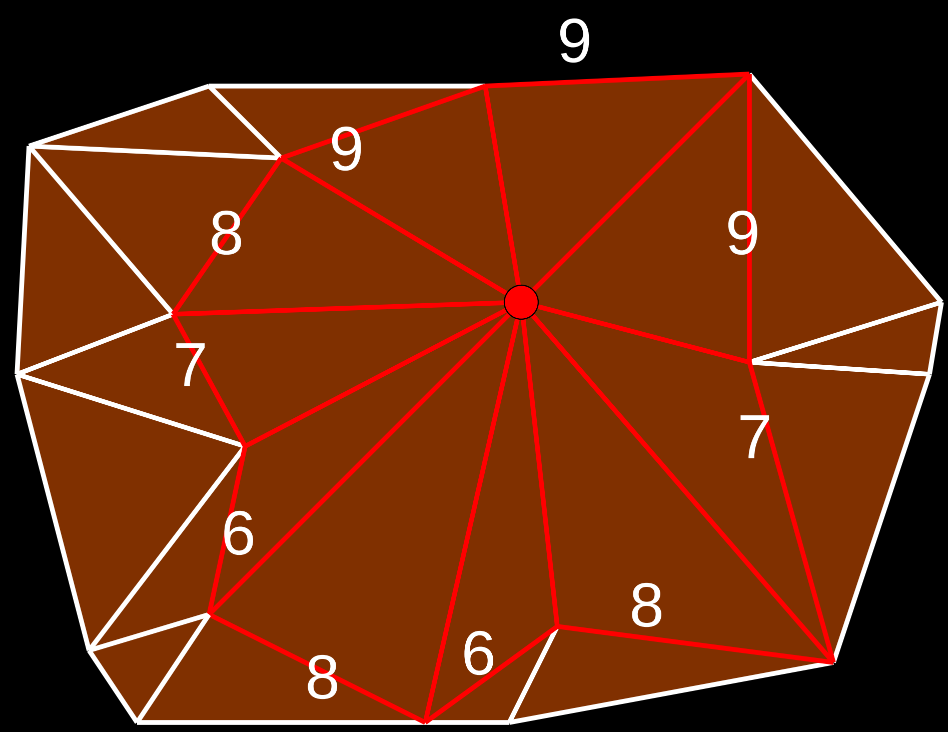
Goal: Find cut between root and leaves that maximizes the smallest cut edge.

Vertex Insertion



Goal: Find cut between root and leaves that maximizes the smallest cut edge.

Vertex Insertion



Scheduling the Transformations

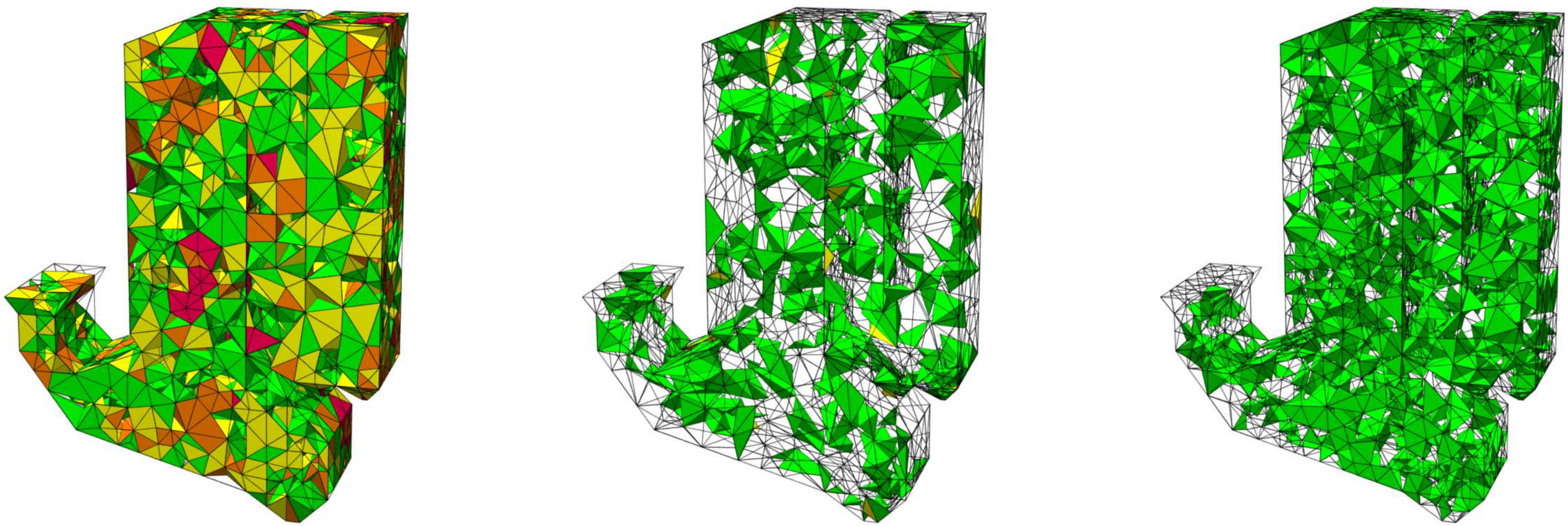
- <10° or >165°
- <20° or >150°
- <30° or >135°
- <40° or >120°

Results

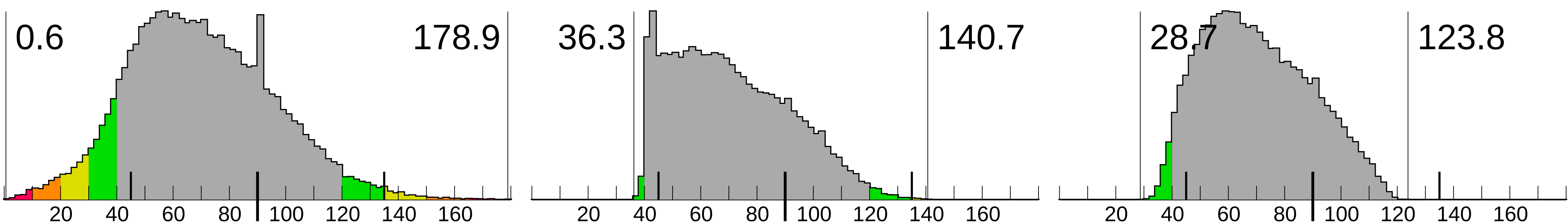
Meshes Improved

<10° or >165°
<20° or >150°
<30° or >135°
<40° or >120°

Tire incinerator | min sine: 163 s | V/l³: 437 s



11,099 tets | 7,066 tets | 10,886 tets

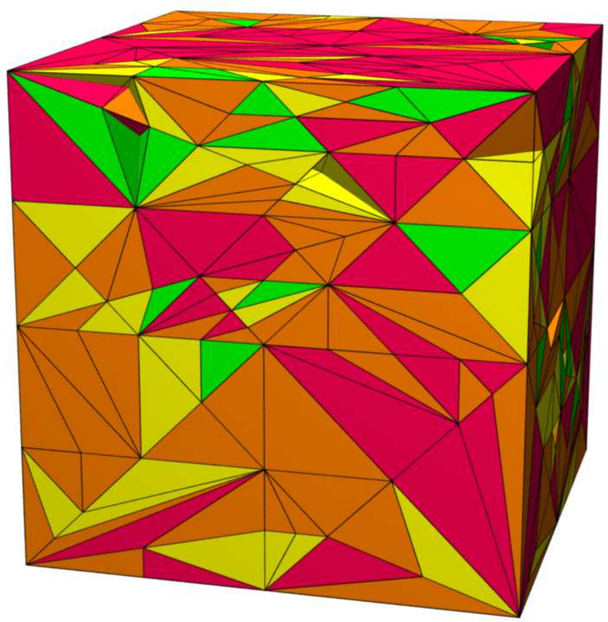


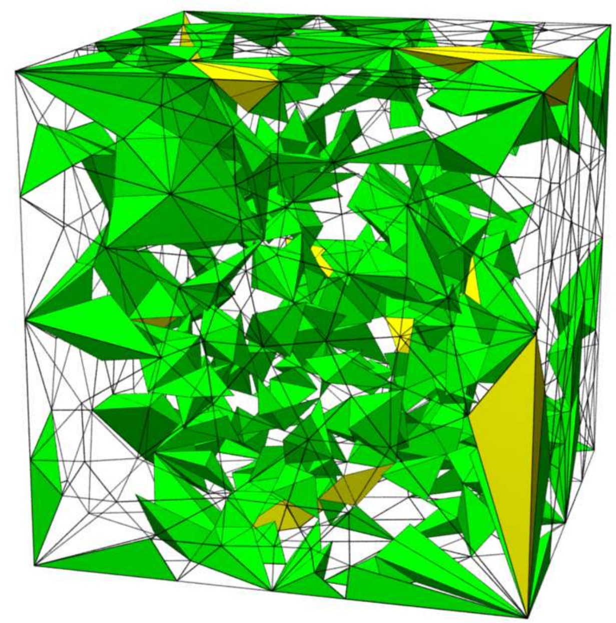
FOG ’97: 13.67° minimum, 156.14° maximum.

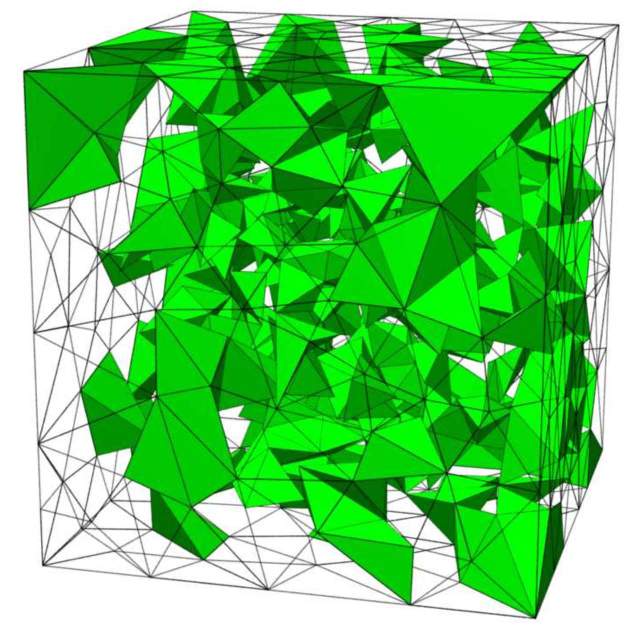
Meshes Improved

<10° or >165°
<20° or >150°
<30° or >135°
<40° or >120°

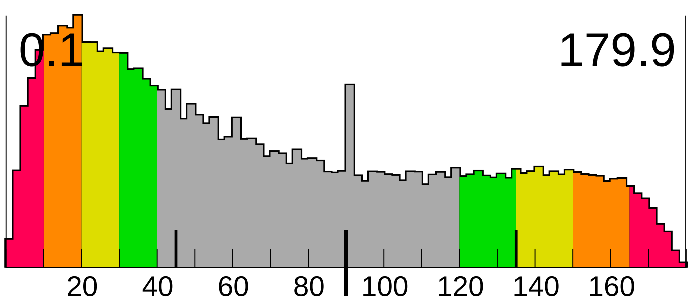
Random mesh	min sine: 234 s	V/l³: 102 s

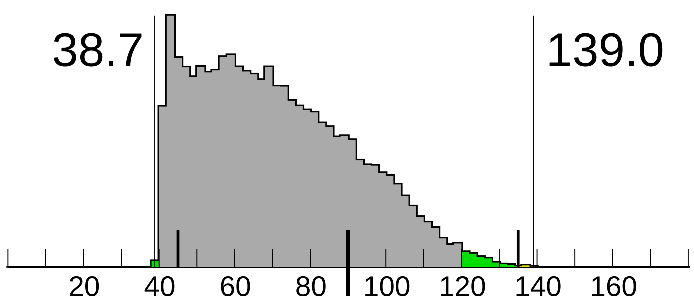


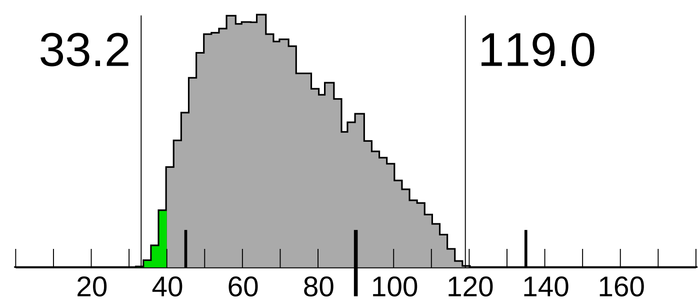




25,705 tets	5,025 tets	3,730 tets





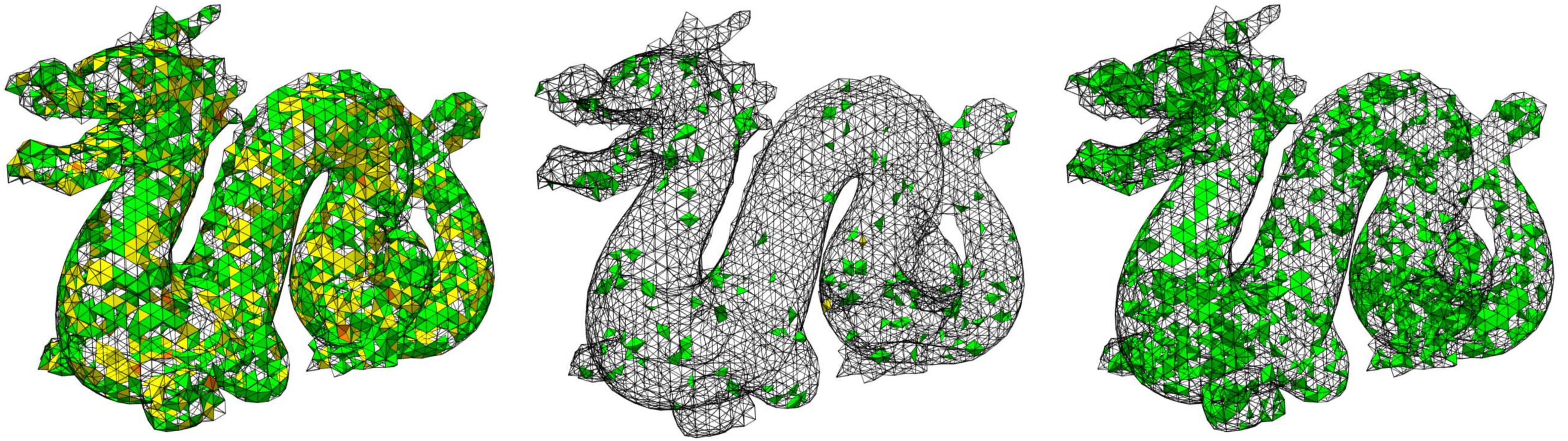


FOG '97: 10.58° minimum, 164.09° maximum.

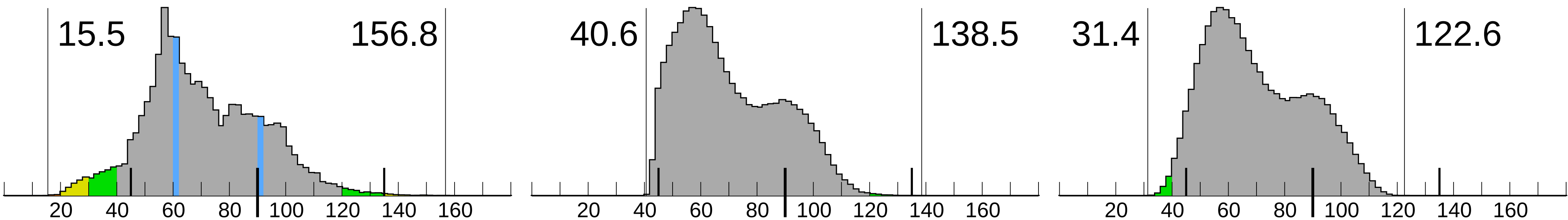
Meshes Improved

<10° or >165°
<20° or >150°
<30° or >135°
<40° or >120°

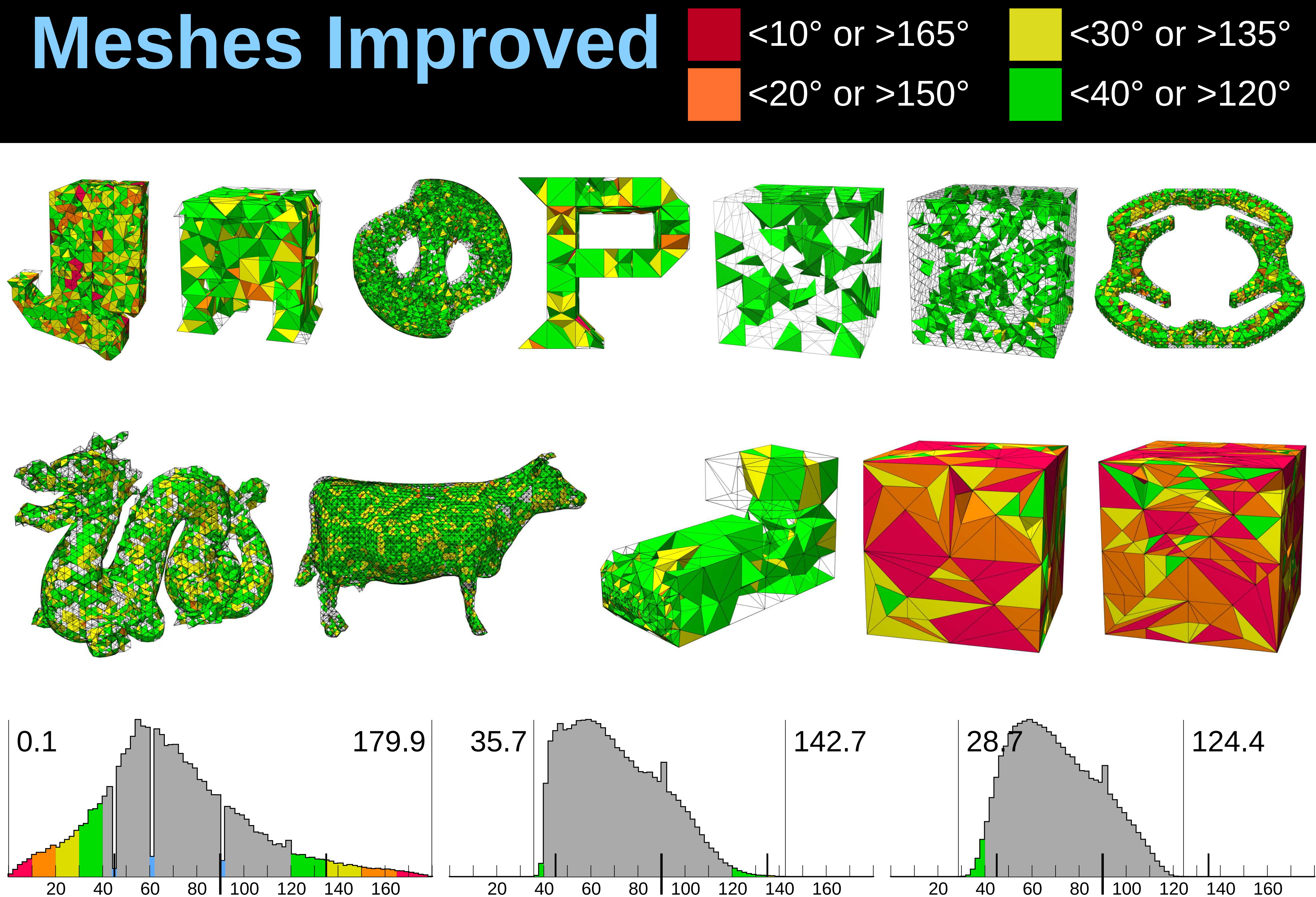
Stanford Dragon | min sine: 747 s | V/l^3: 752 s



32,960 tets | 34,672 tets | 35,389 tets

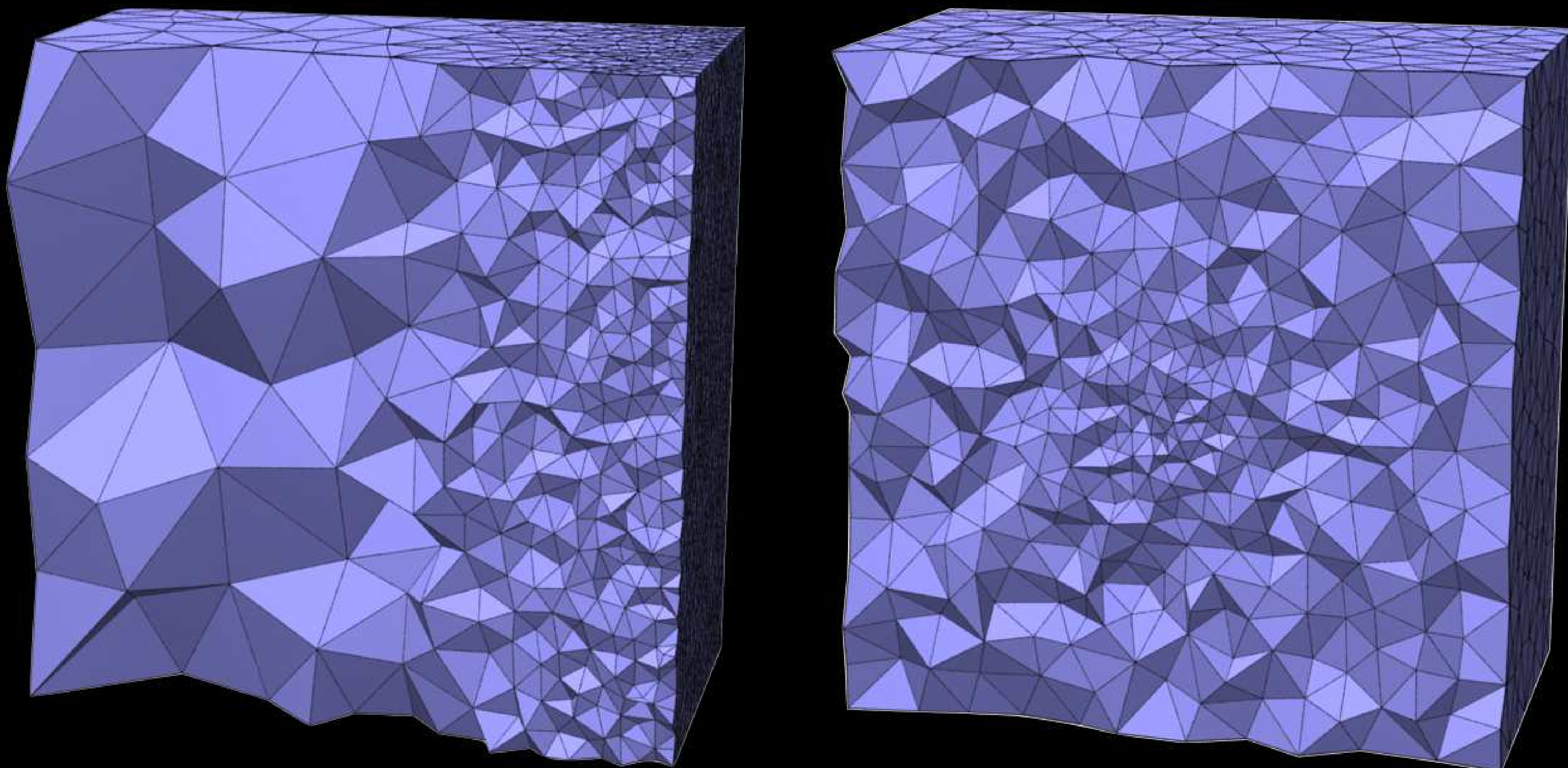


Meshes Improved

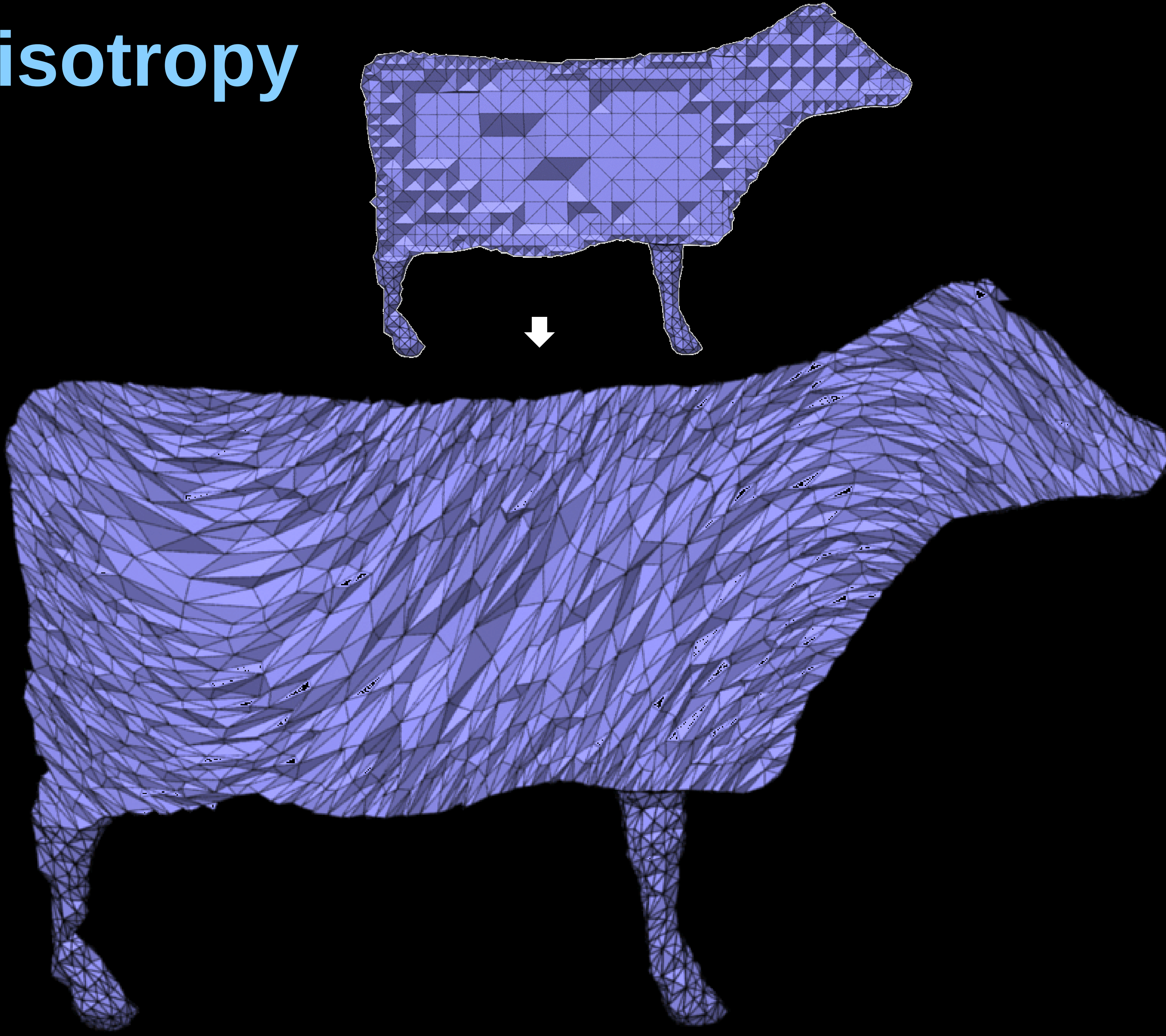


12 input meshes minimum sine V/l³

Size Control

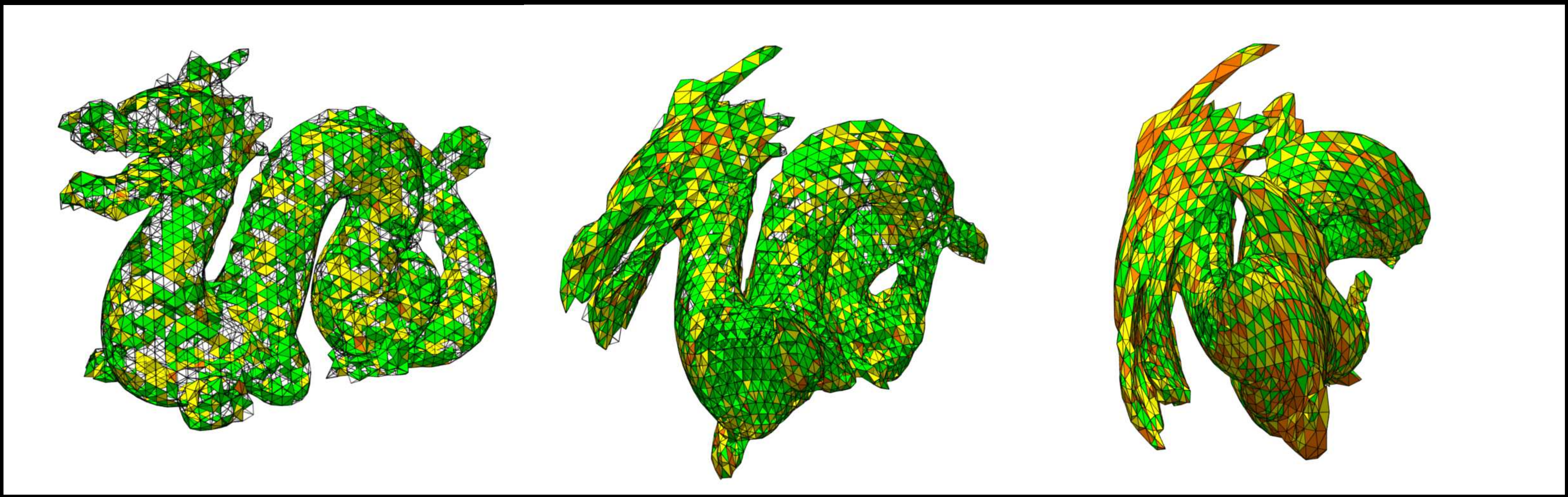


Anisotropy

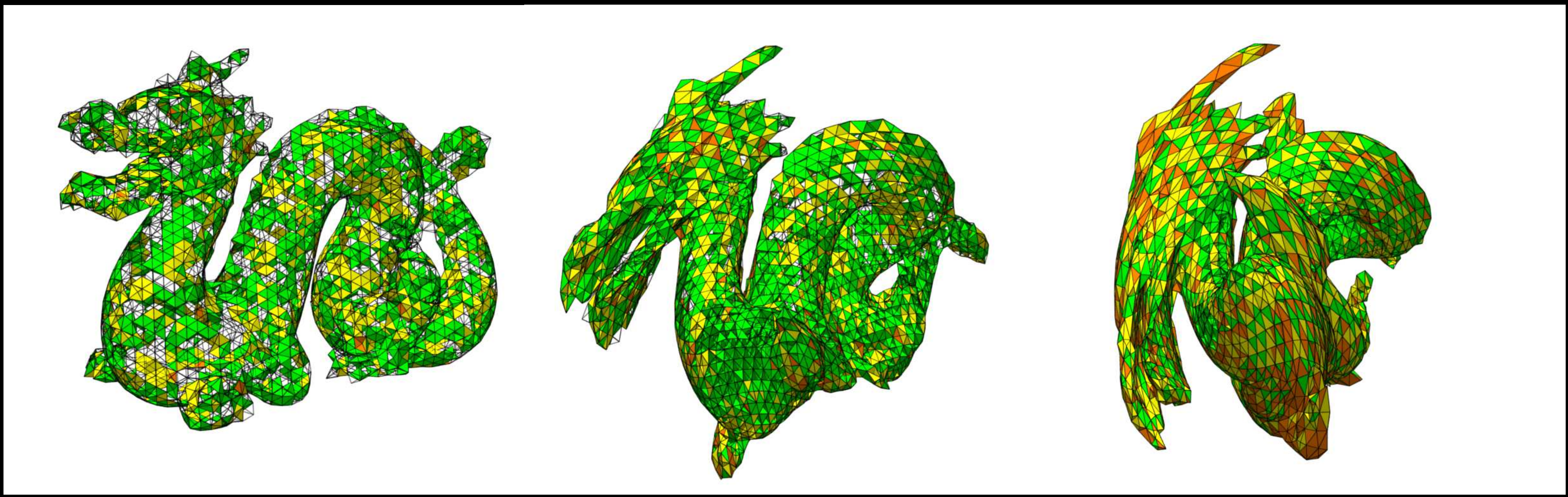


Dynamic Meshing

Dynamic Meshing

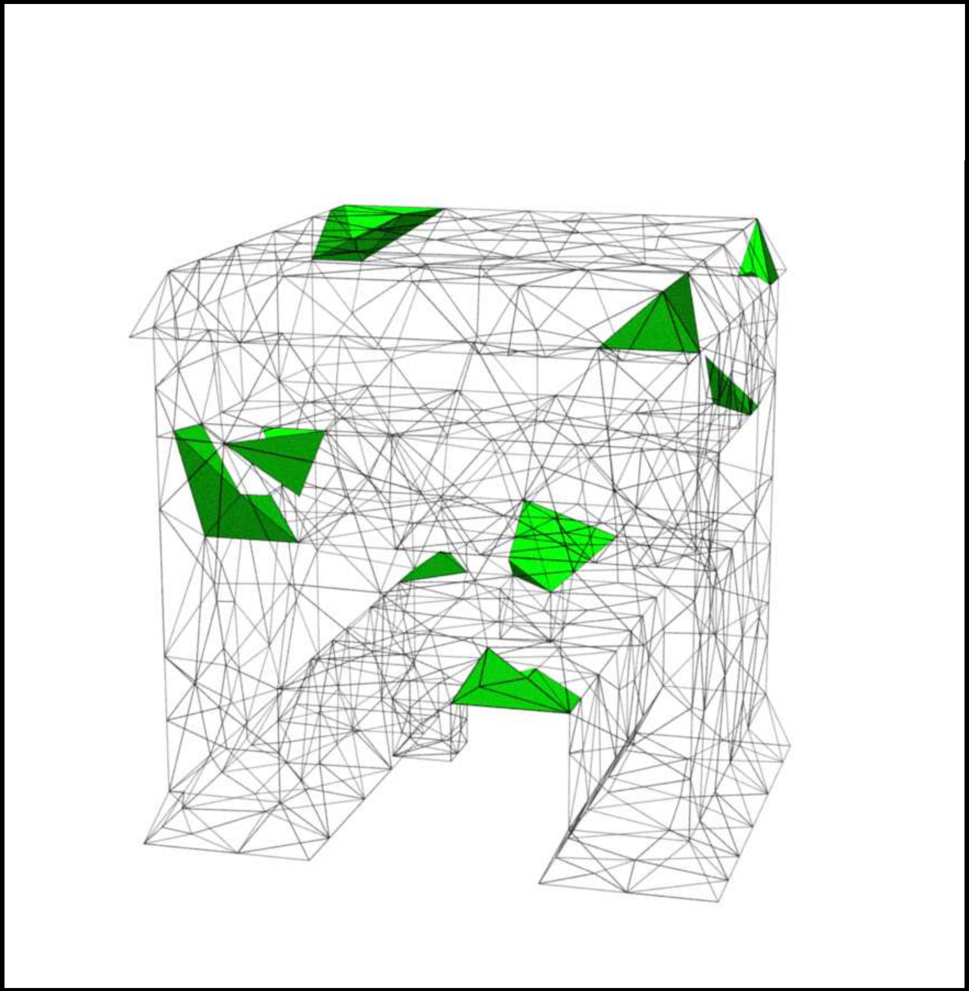


Dynamic Meshing

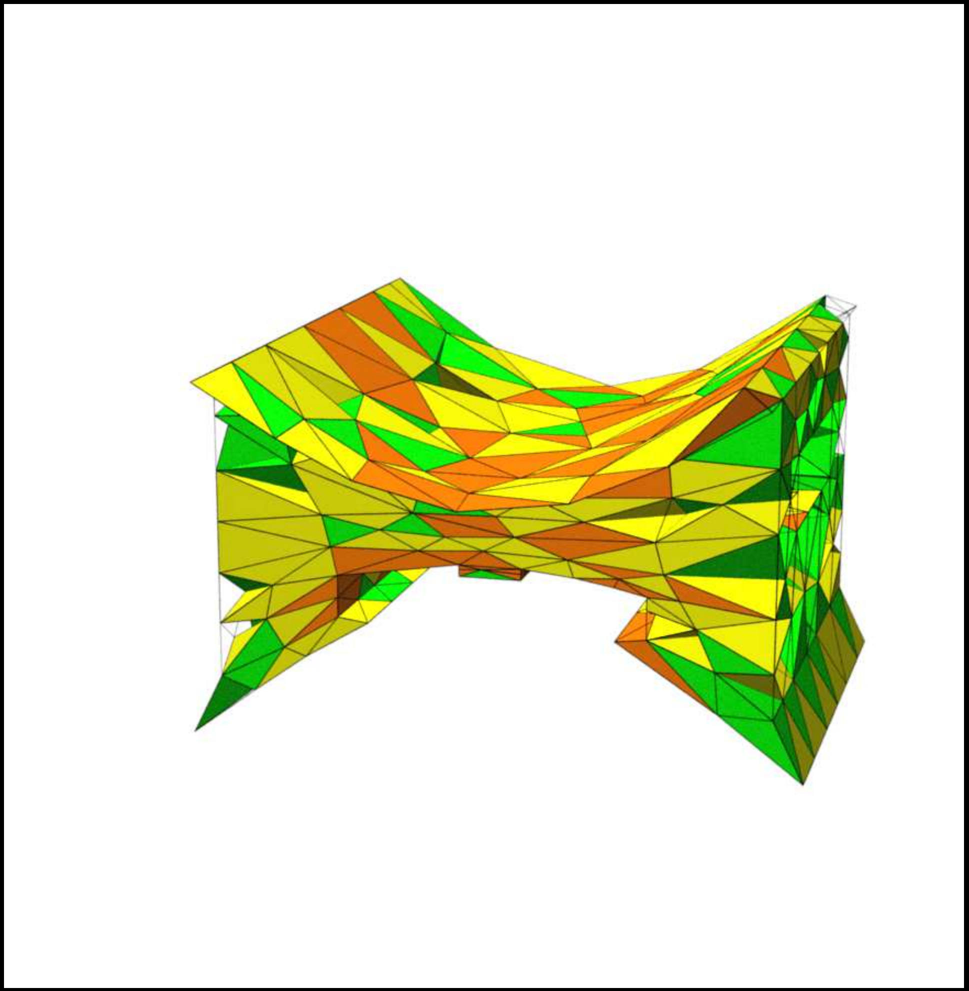


Motivation: accuracy, not speed.

Mesh Improvement

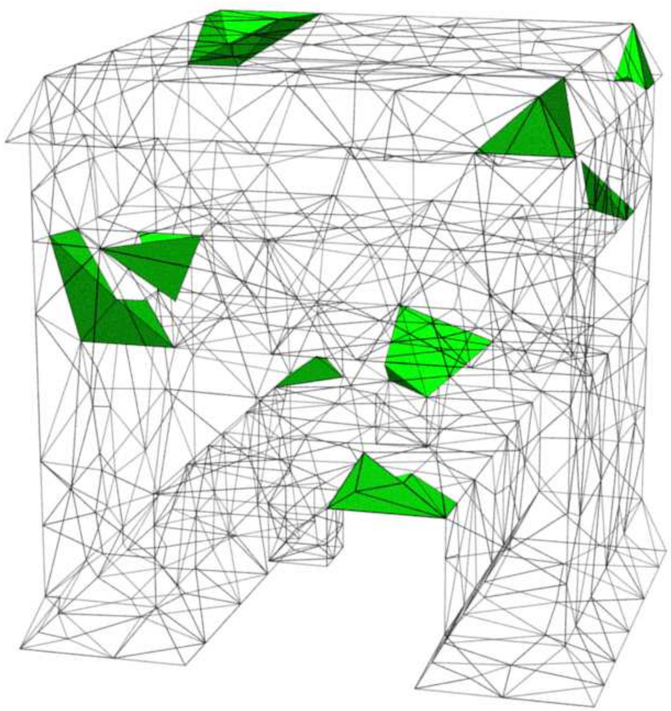


Dynamic Meshing



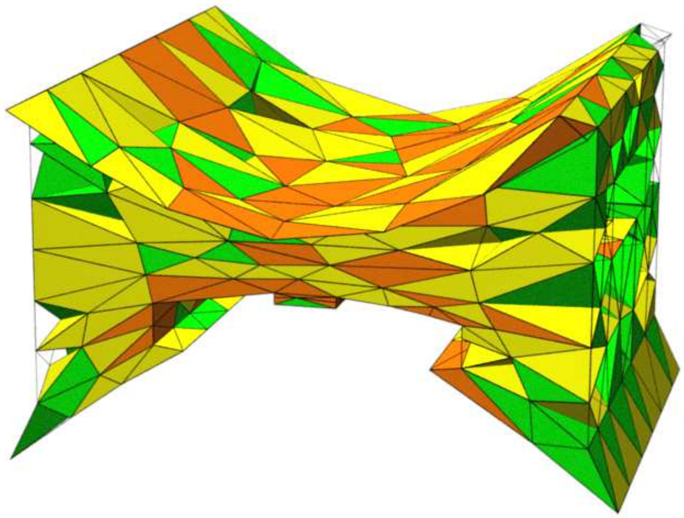
Mesh Improvement

Goal: Maximize quality



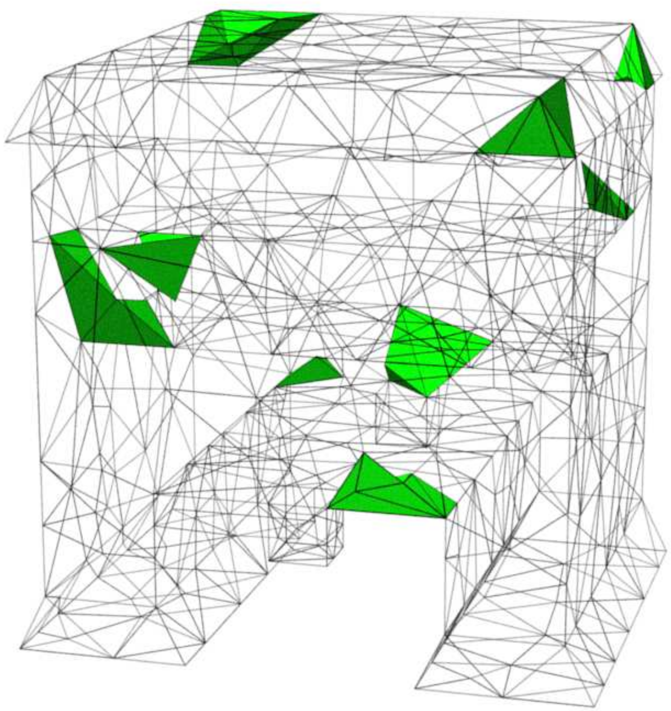
Dynamic Meshing

Goal: Change as little as possible



Mesh Improvement

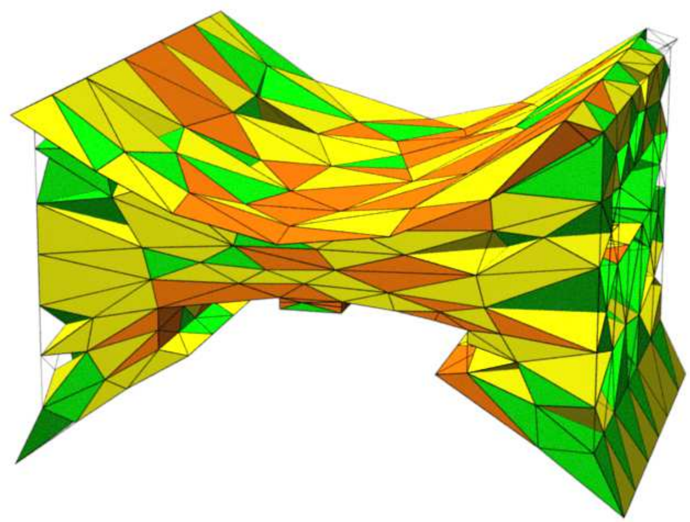
Goal: Maximize quality



Try to improve every tet.

Dynamic Meshing

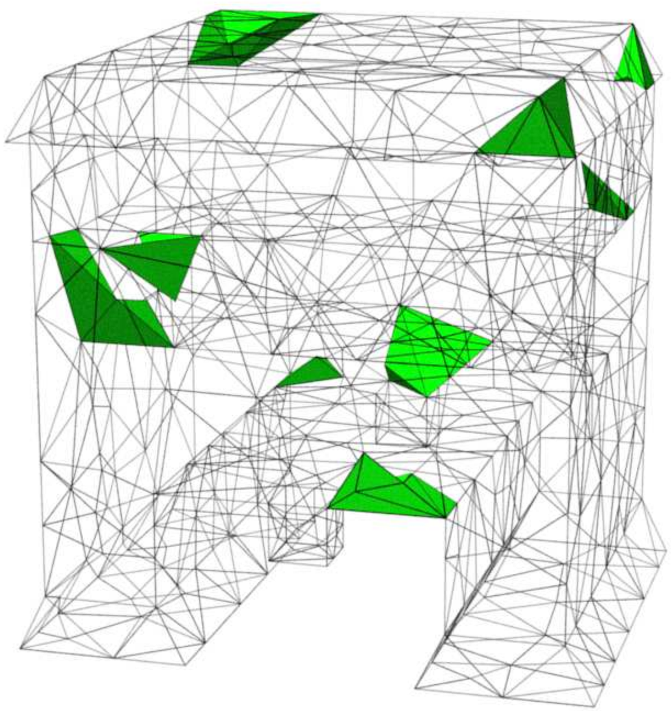
Goal: Change as little as possible



Target only poor tets.

Mesh Improvement

Goal: Maximize quality

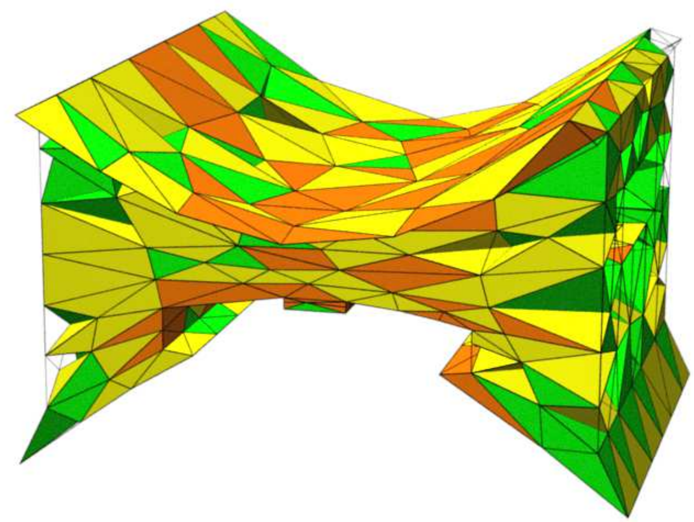


Try to improve every tet.

Try fastest operations first.

Dynamic Meshing

Goal: Change as little as possible



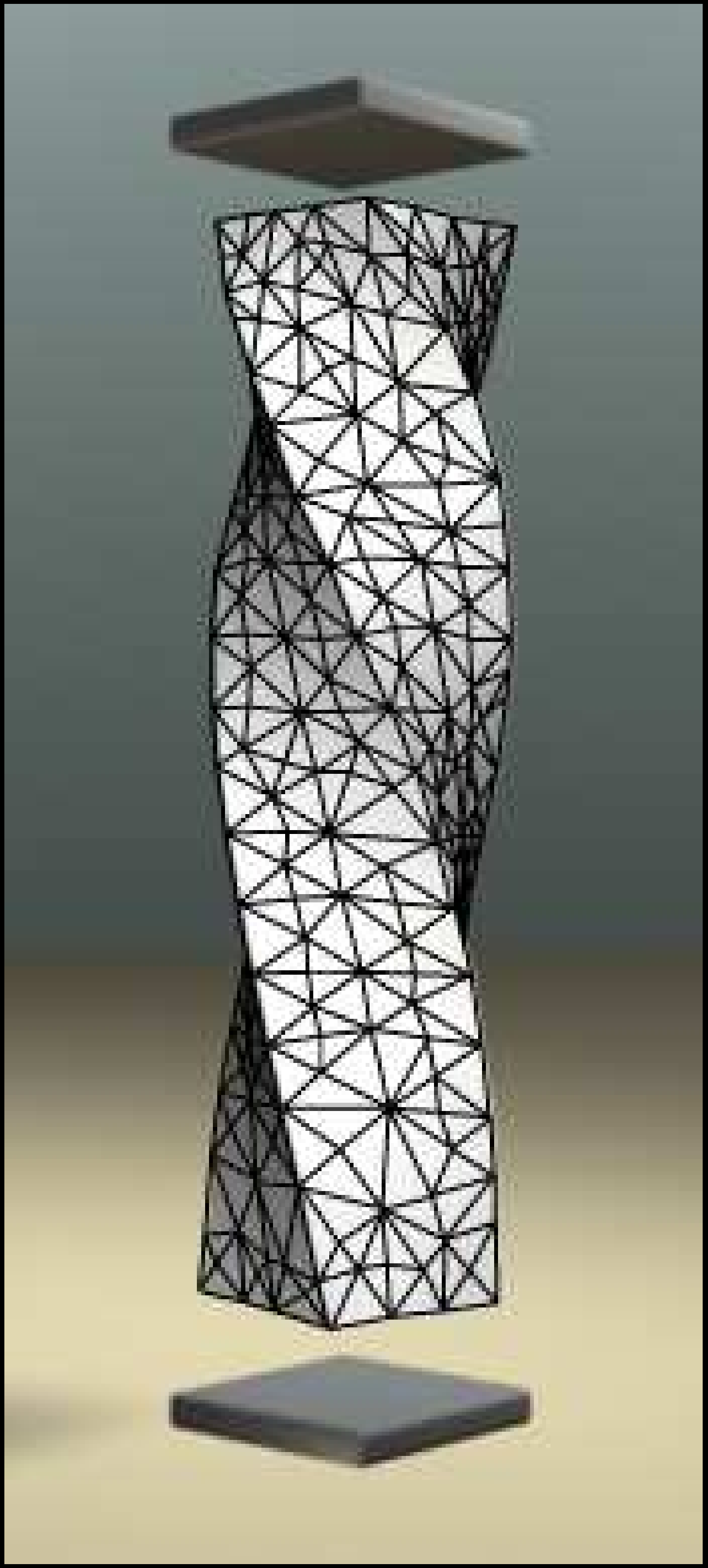
Target only poor tets.

Try most conservative first.

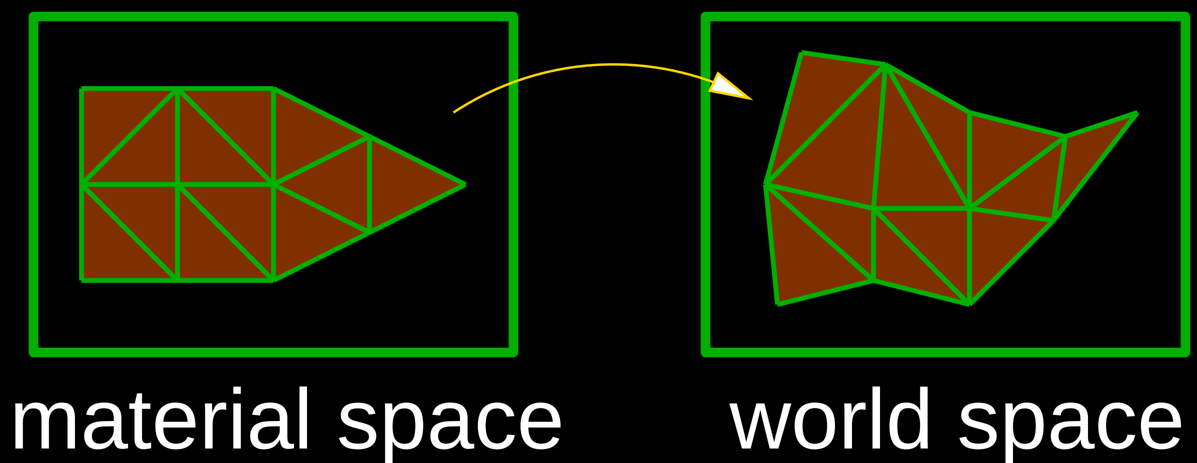
Dynamic Local Remeshing for Elastoplastic Simulation

Martin Wicke
Daniel Ritchie
Bryan Klingner
Sebastian Burke
Jonathan Shewchuk
James O’Brien

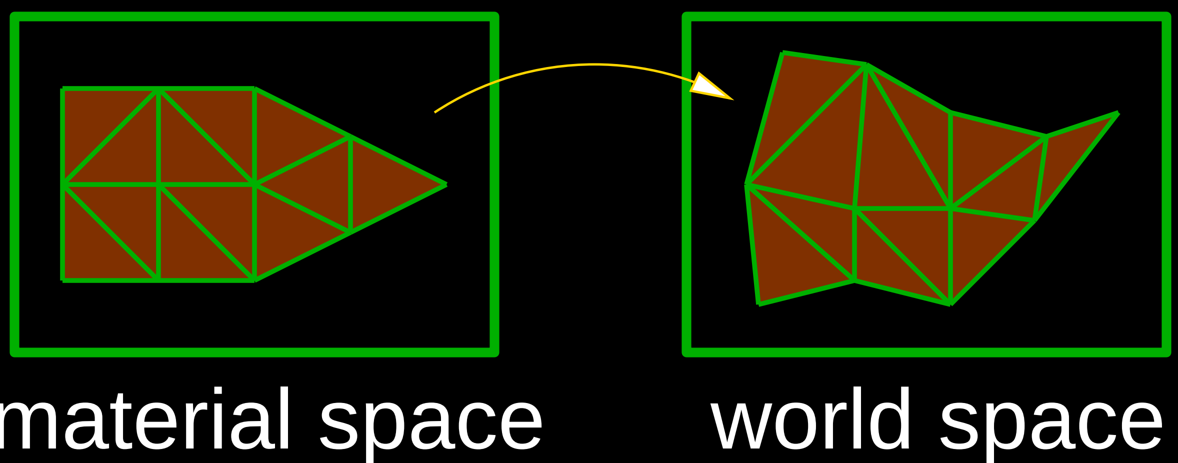
Computer Science Division
University of California
Berkeley, California



Lagrangian Finite Elements



Lagrangian Finite Elements

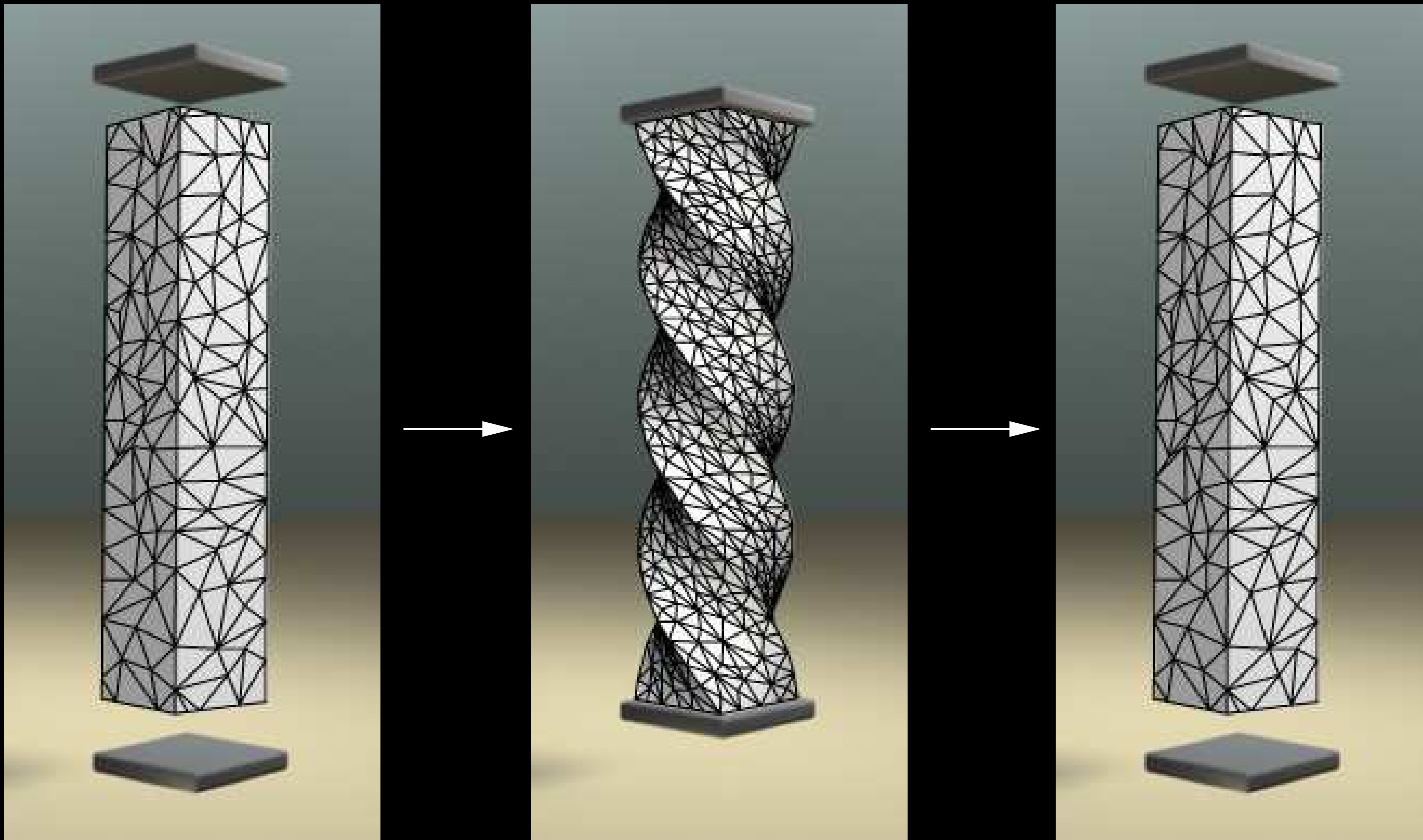


Strains are inferred from material–world mapping.

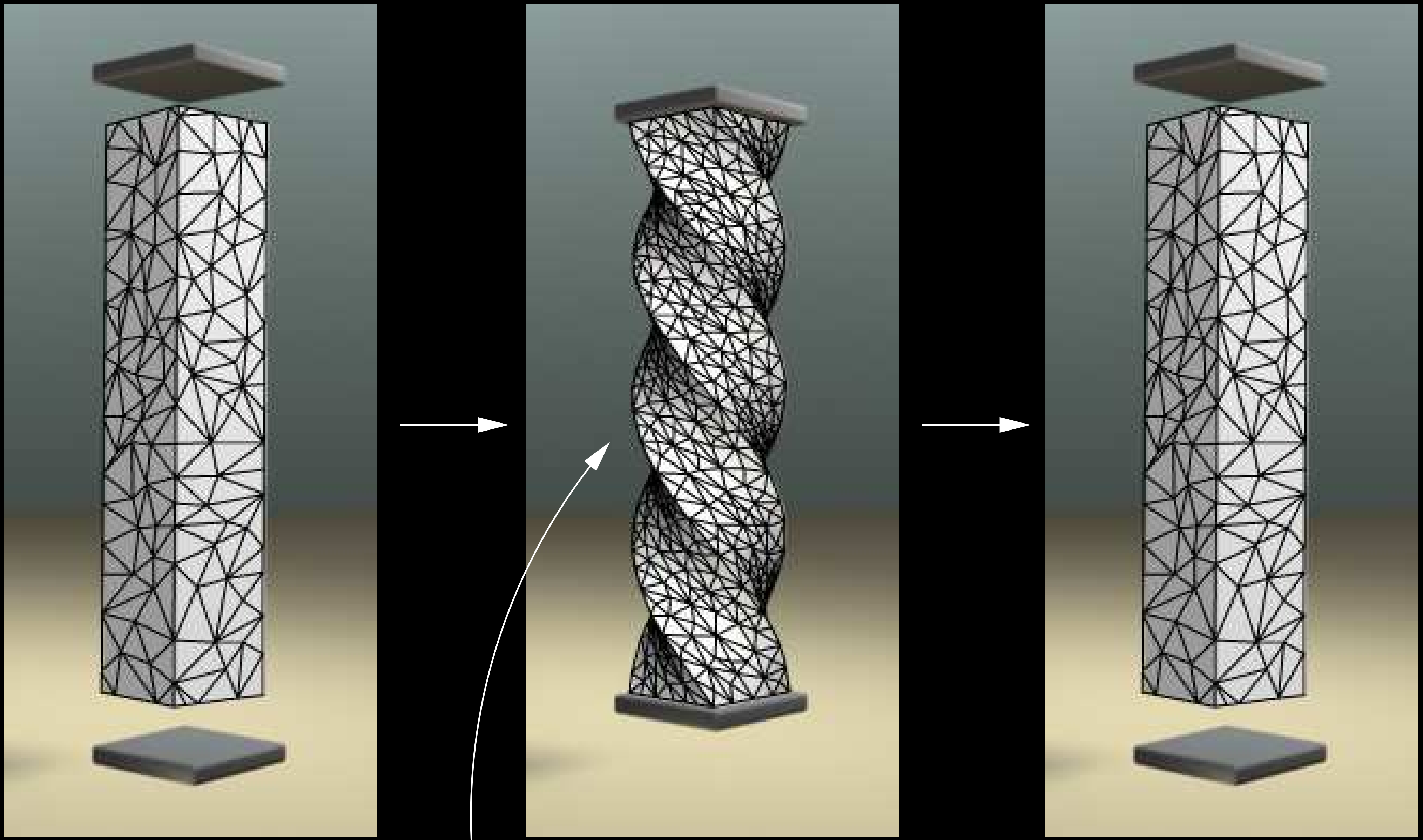
Forces/stresses are inferred from strains.

Motion/acceleration is inferred from forces.

Elastic Deformation

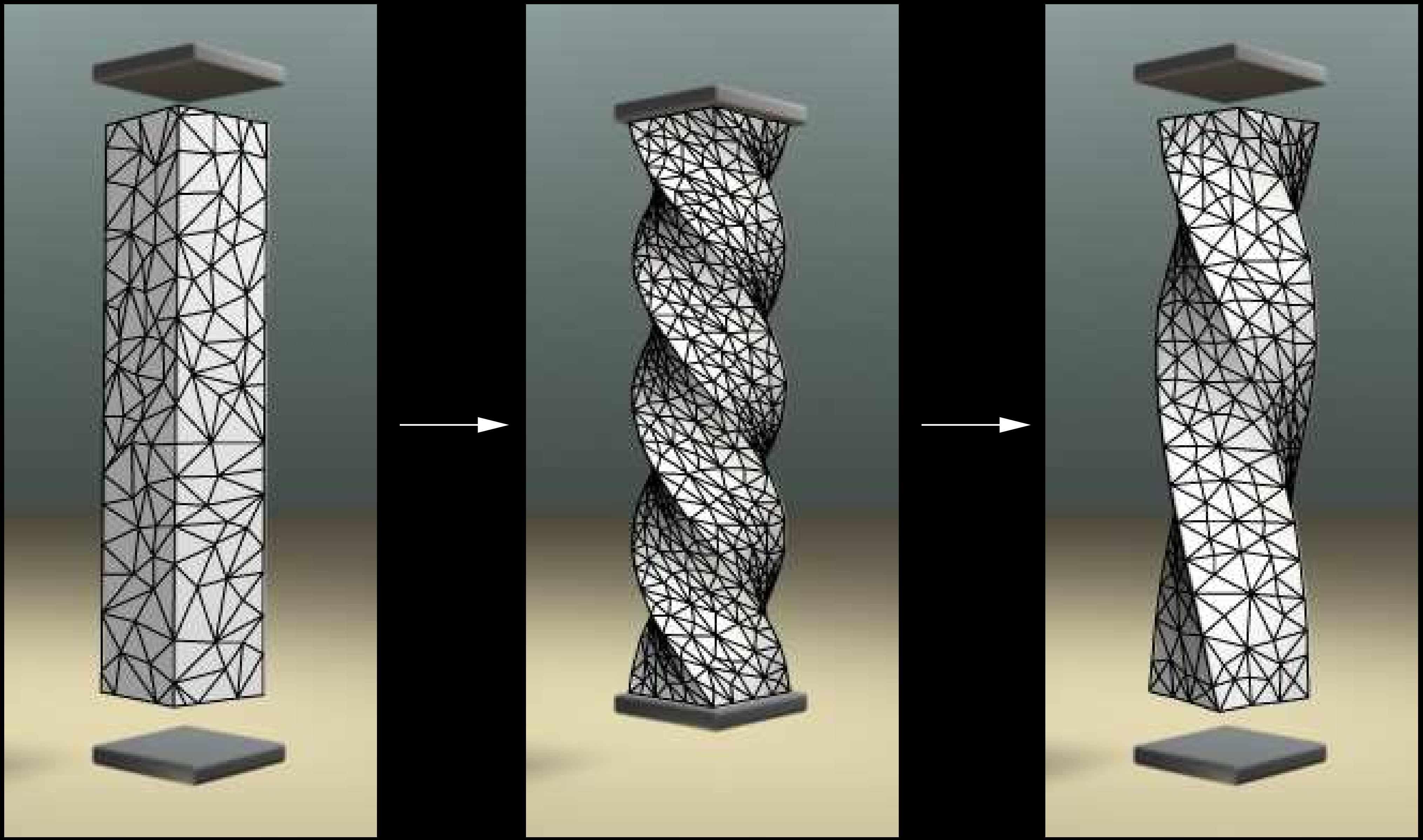


Elastic Deformation

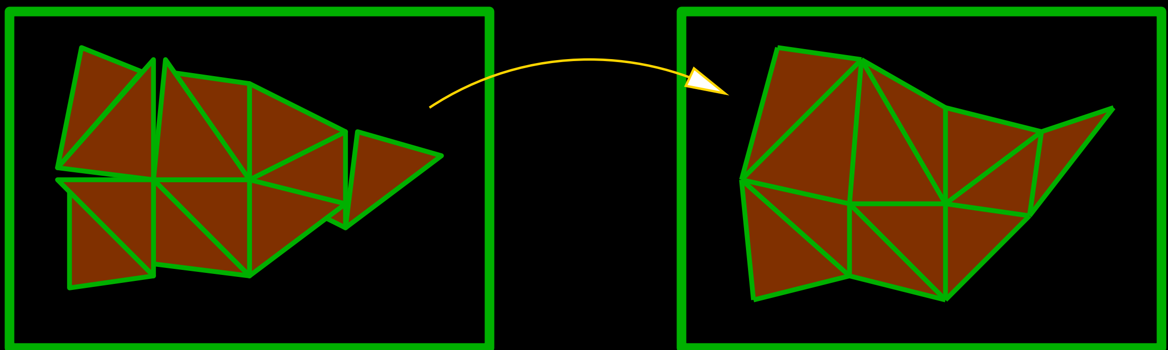


Remeshed in material space

Plastic Deformation



Plastic Flow: No Material Space

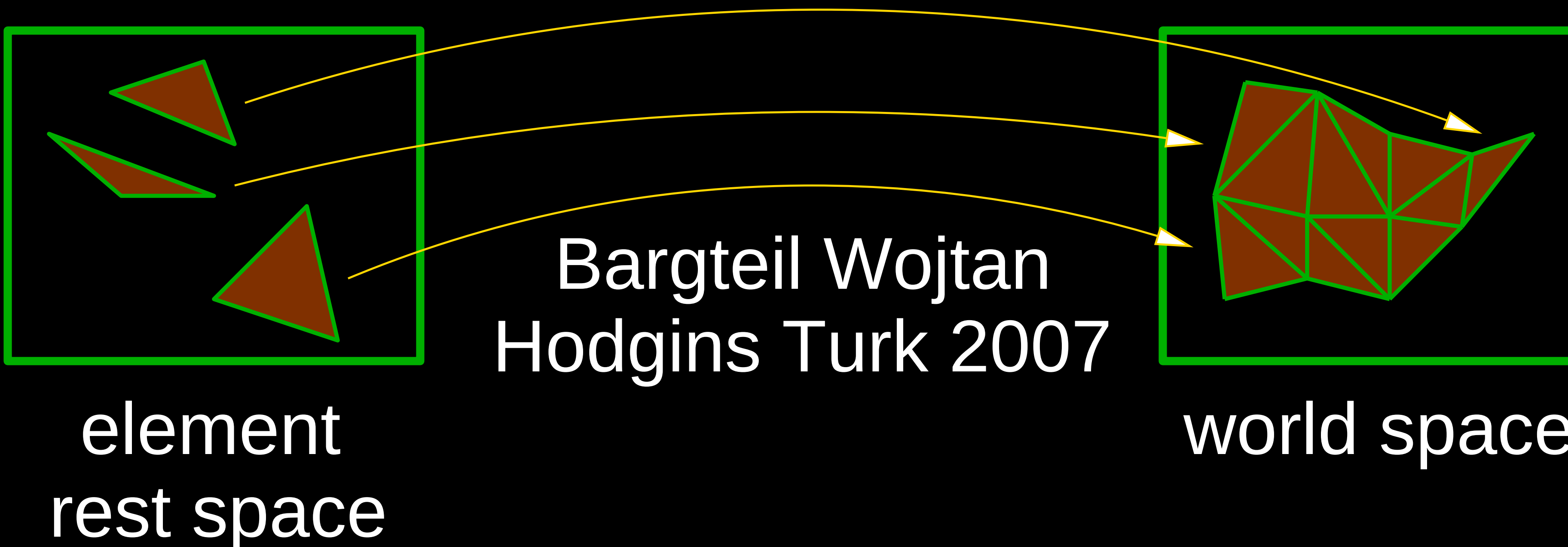


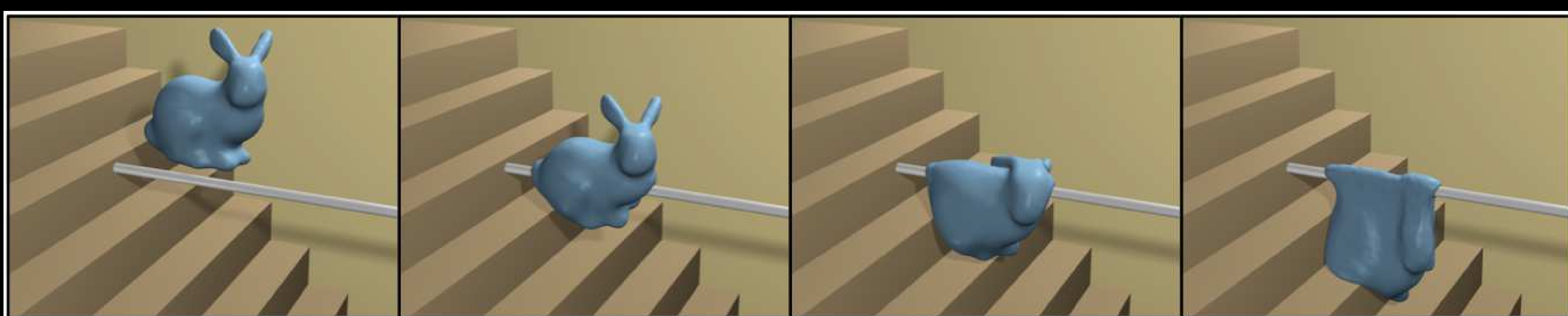
material space??   world space

Elements' rest shapes change permanently.

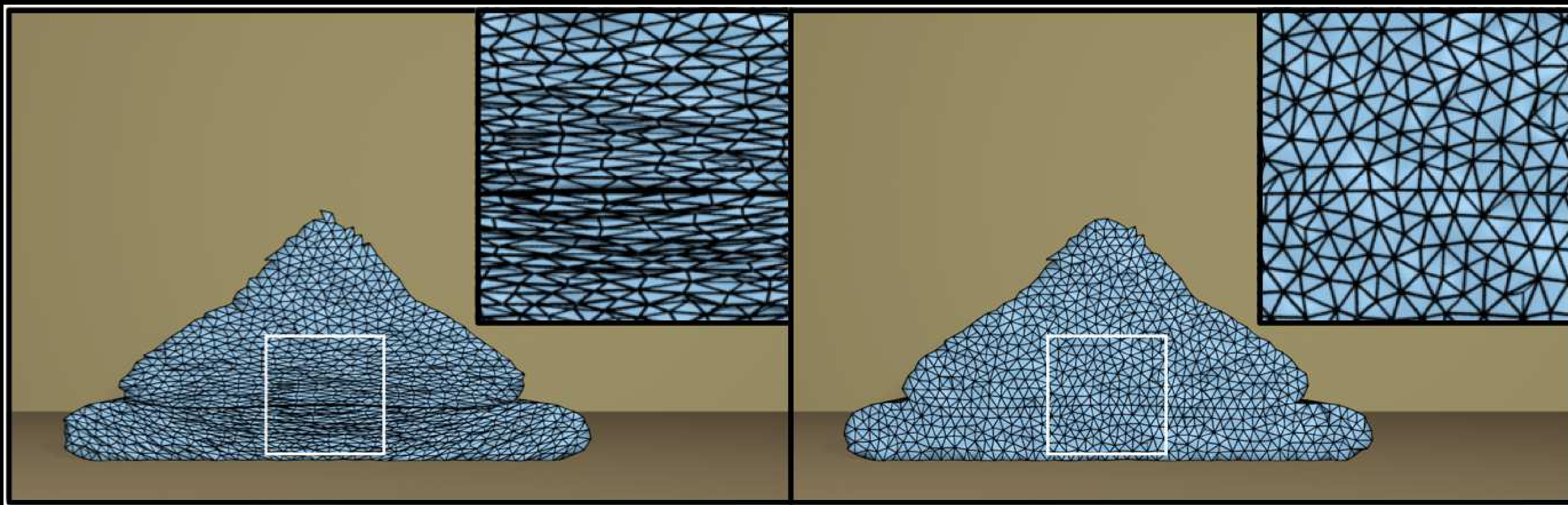
They no longer fit together.

Solution: No Material Space

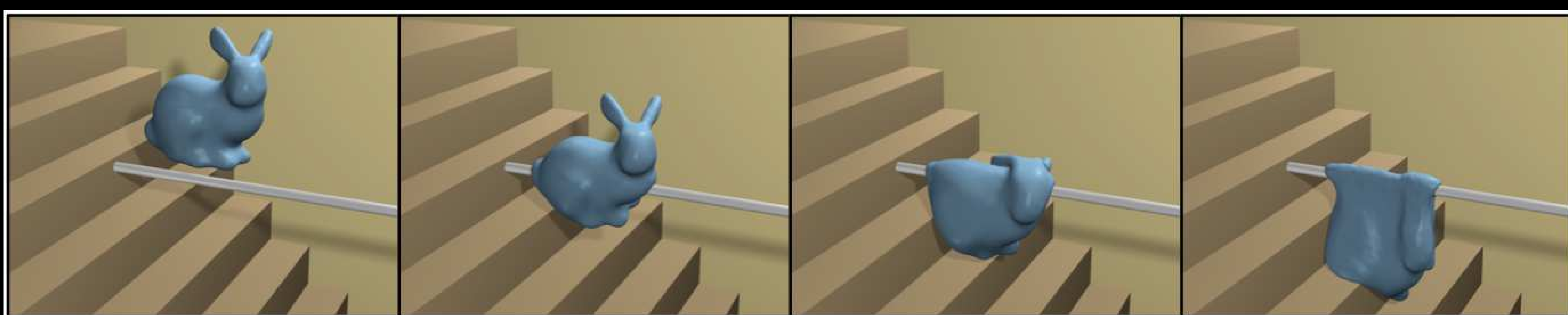




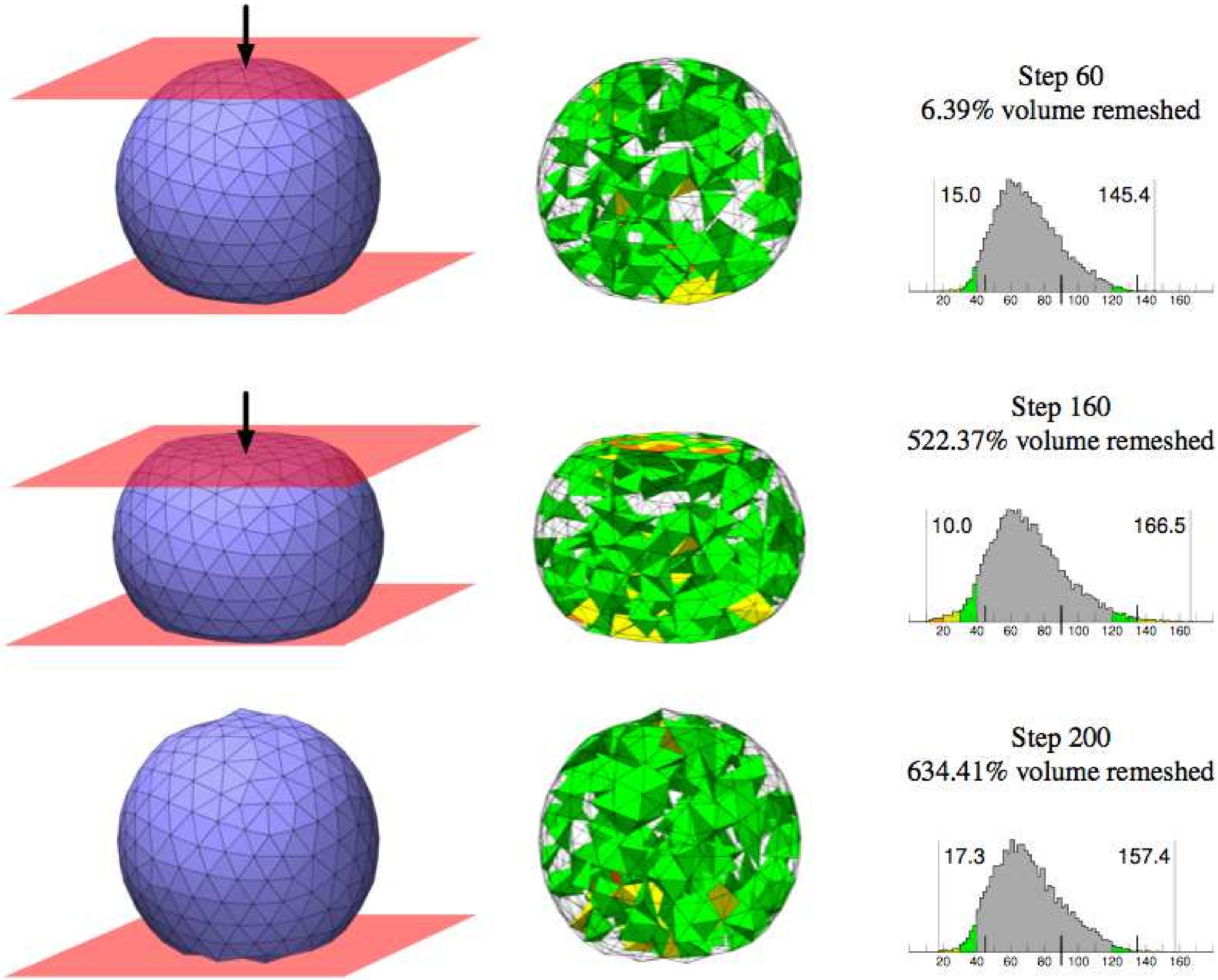
Solution: No Material Space



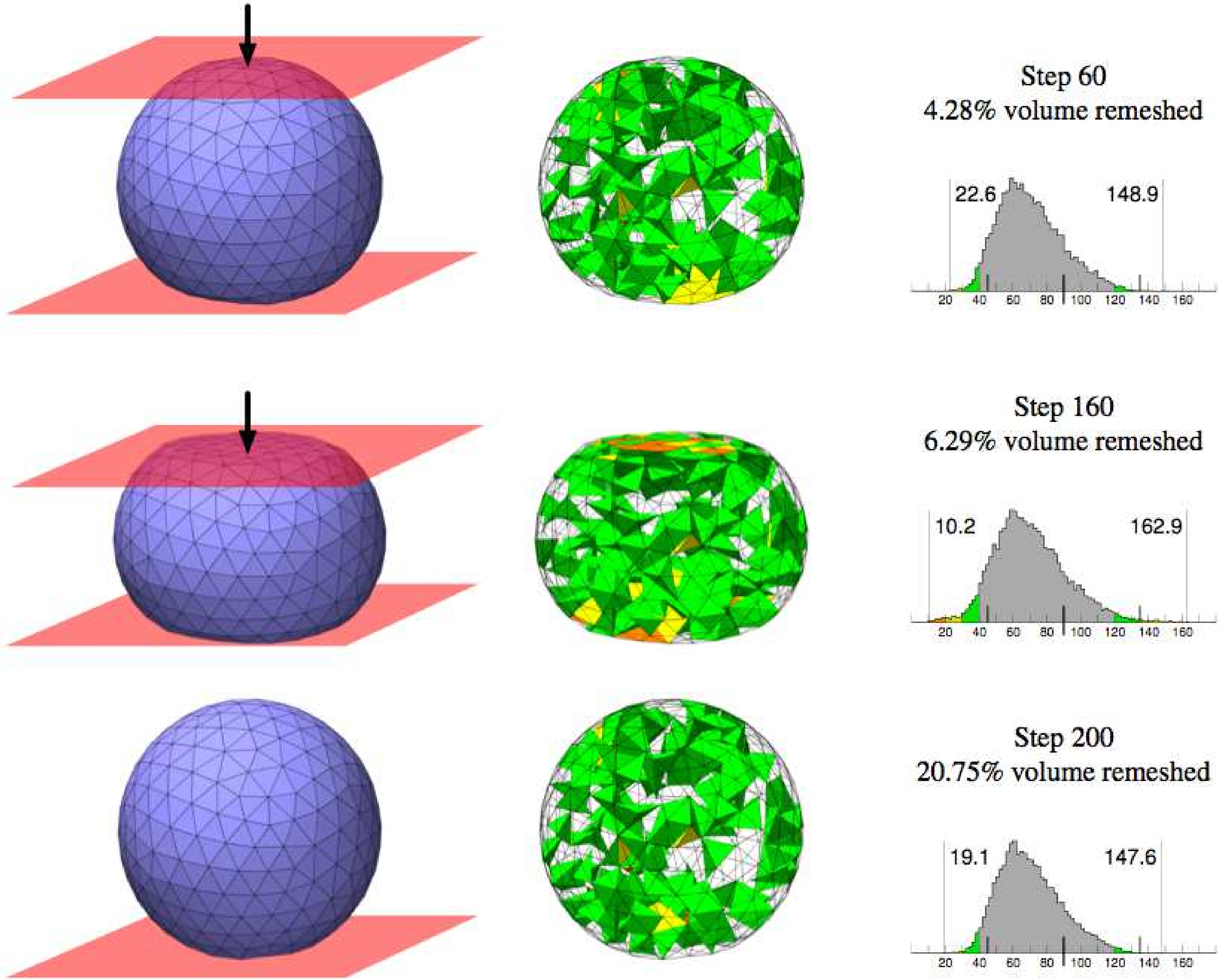
Bargteil/Wojtan/Hodgins/Turk remesh in world space.



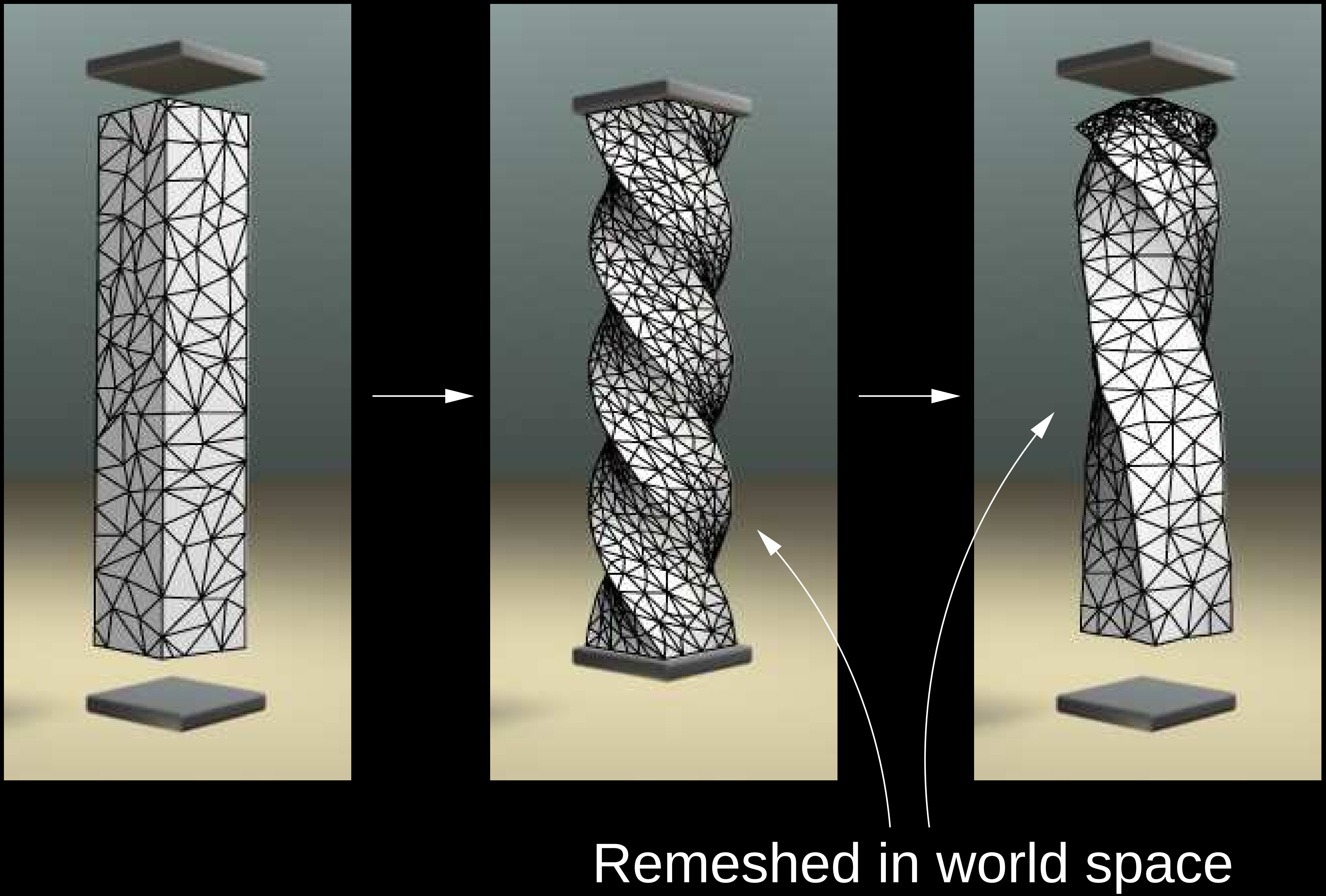
Remeshing ⇒ Artificial Diffusion



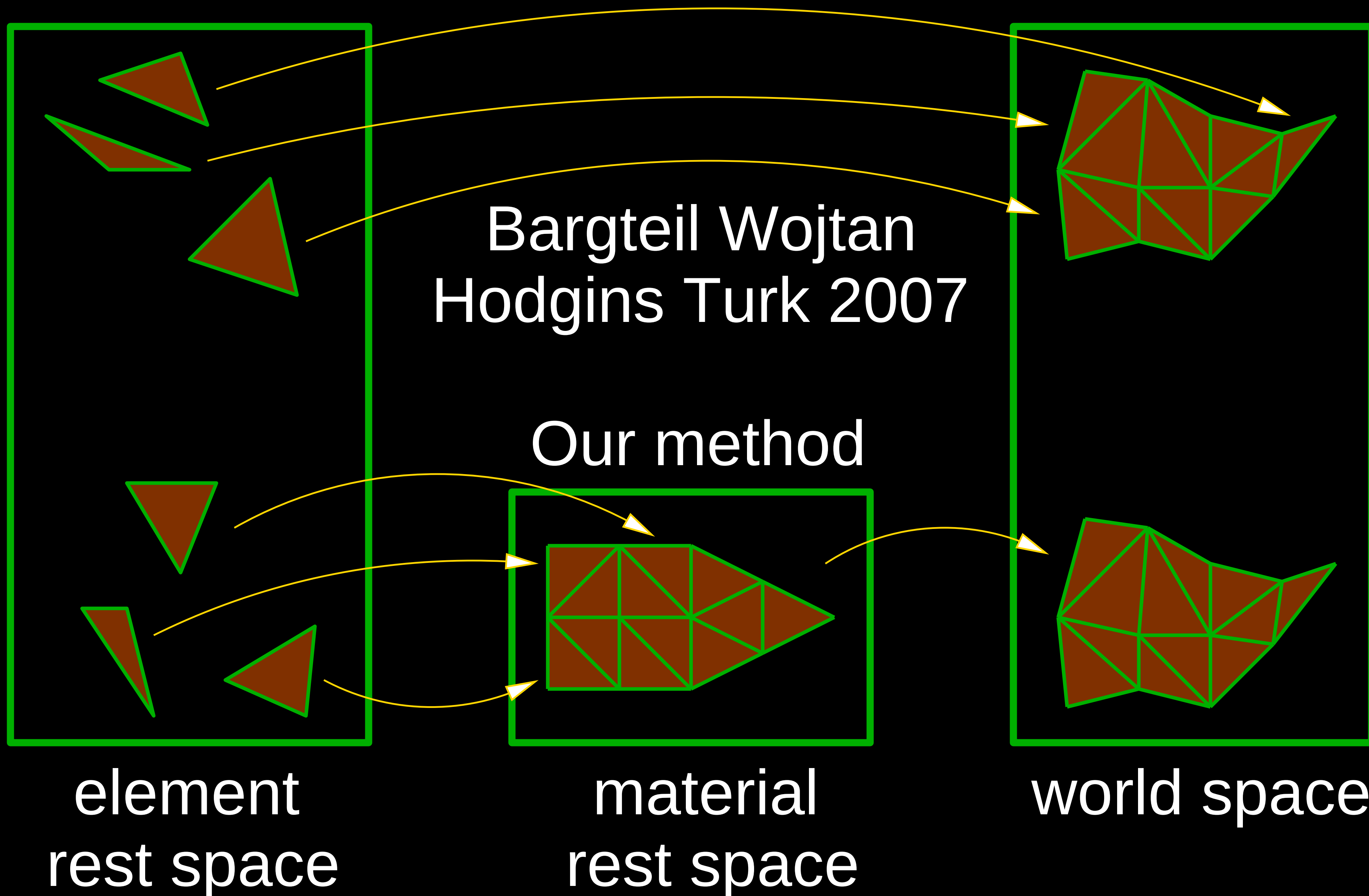
Dynamic Meshing ⇒ Less Error



Elasticity & Artificial Diffusion



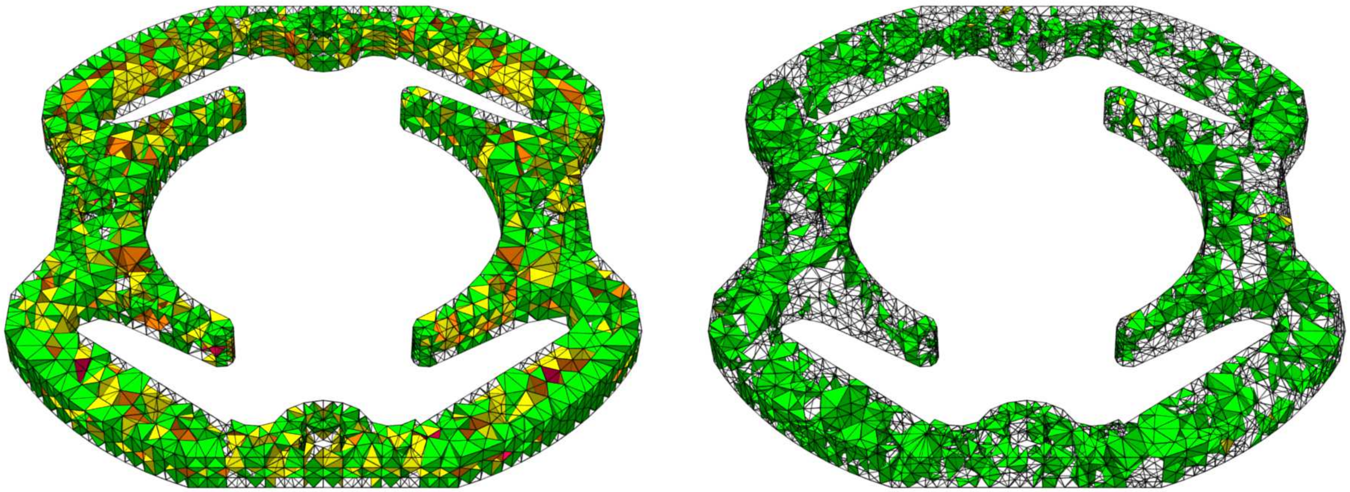
Plasticity with 3 Spaces



Conclusions

Conclusions

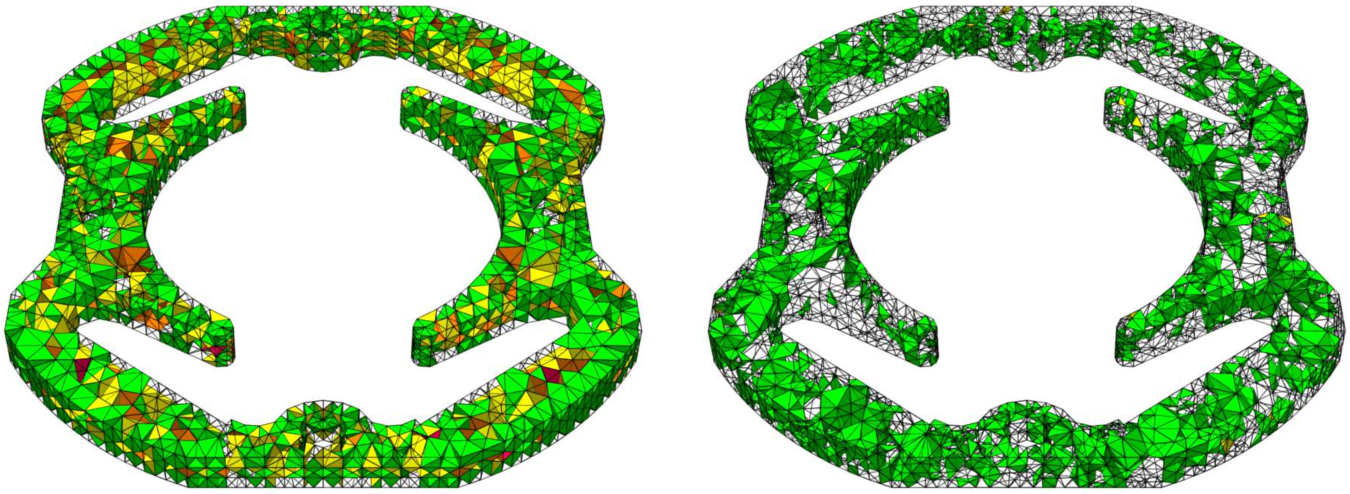
In practice, usually achieves meshes of **far** higher quality than those obtained by any other algorithm for mesh generation or improvement.



Conclusions

In practice, usually achieves meshes of **far** higher quality than those obtained by any other algorithm for mesh generation or improvement.

Consistent enough for dynamic meshing.

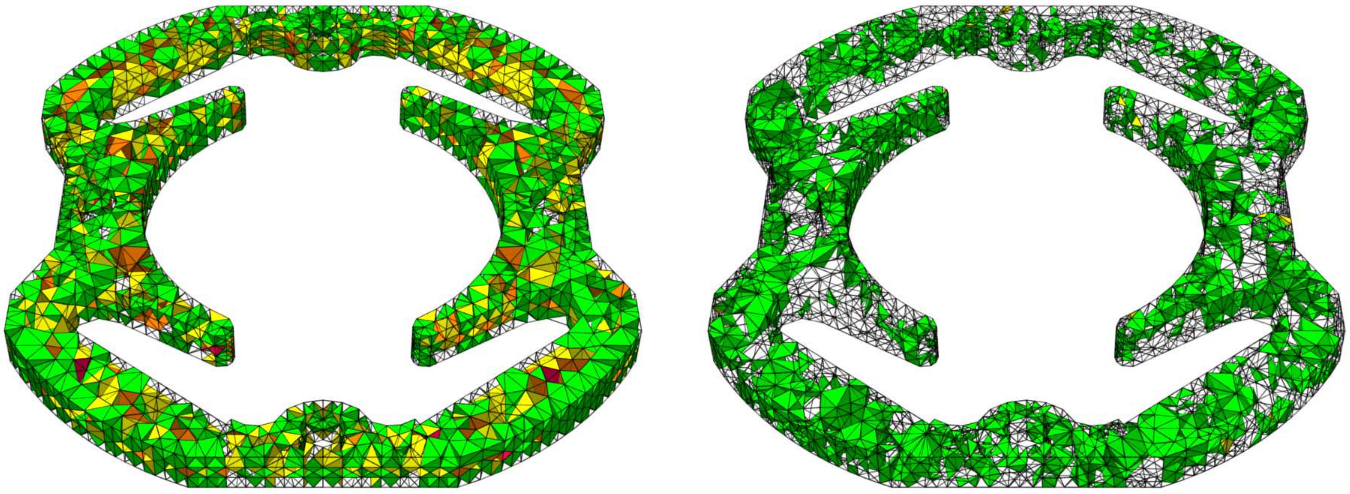


Conclusions

In practice, usually achieves meshes of **far** higher quality than those obtained by any other algorithm for mesh generation or improvement.

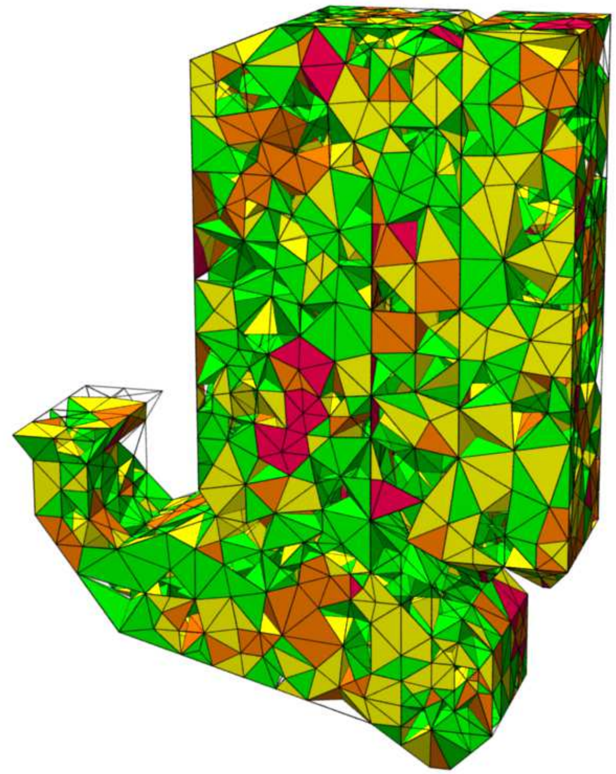
Consistent enough for dynamic meshing.

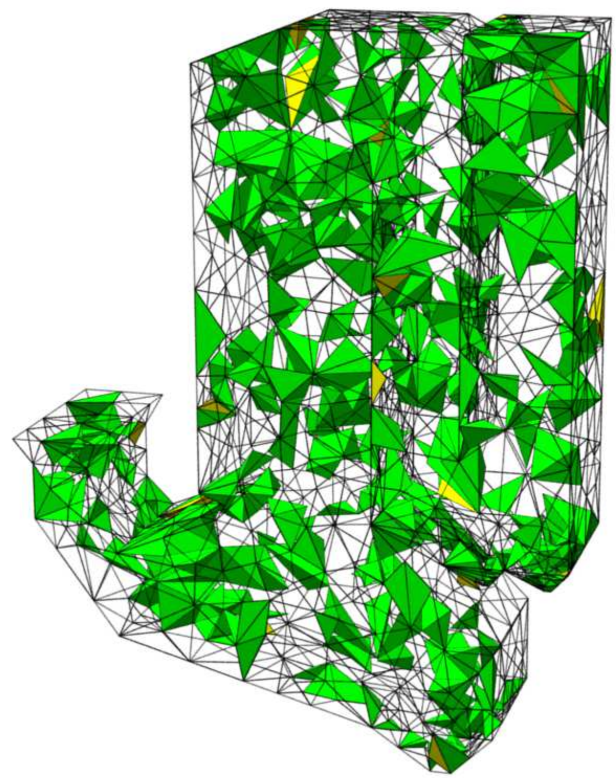
Makes possible simulations with light plastic flow, or a wide range within one object.



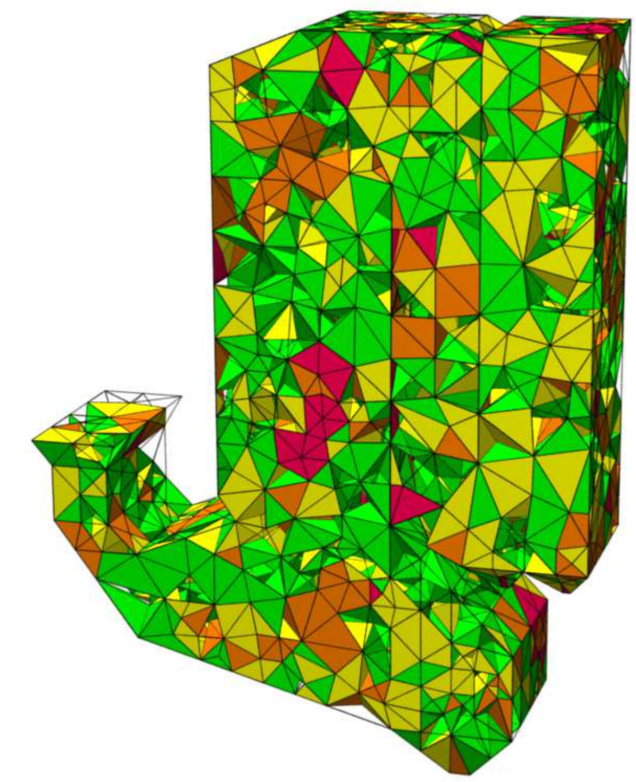
Bryan Matthew Klingner and Jonathan Richard Shewchuk, "Aggressive Tetrahedral Mesh Improvement," Proceedings of the 16th Annual Meshing Roundtable, pages 3−23, October 2007.

Bryan Matthew Klingner, "Improving Tetrahedral Meshes," Ph.D. dissertation, Technical Report UCB/EECS−2008−145, Dept. of EECS, UC Berkeley, 2008.





Bryan Matthew Klingner and Jonathan Richard Shewchuk, "Aggressive Tetrahedral Mesh Improvement," Proceedings of the 16th Annual Meshing Roundtable, pages 3−23, October 2007.



http://www.cs.berkeley.edu/~jrs/stellar

Bryan Matthew Klingner, "Improving Tetrahedral Meshes," Ph.D. dissertation, Technical Report UCB/EECS−2008−145, Dept. of EECS, UC Berkeley, 2008.