Surface Simplification Using Quadric Error Metrics

Authors: Michael Garland & Paul Heckbert

Presented by: Niu Xiaozhen

Disclaimer: Some slides are modified from original slides, which were designed by Michael Garland (Author of the paper).

Outline

- **Introduction**
- Motivation
- Background
- Algorithm Summary
- Results and Performance Analysis
- Conclusions

Introduction

- Many graphics applications require complex model to maintain realism.
- However, full complexity of model is not always required.
- Need surface simplification to reduce the computational cost.

Outline

- Introduction
- **Motivation**
- Background
- Algorithm Summary
- Results and Performance Analysis
- Conclusions

Motivation

- The goal of proposed research is focused on:
  - **Efficiency**. How to simplify complex models rapidly?
  - **Quality**. The approximation results should maintain high fidelity to the original model.
  - **Generality**. Do we have to maintain the topology of original model, or not?

Motivation (2)

- Holes and gaps usually disappear in the distance.
- Topology can be simplified.
- Approximation error should be minimized during the simplification.

Outline

- Introduction
- Motivation
- **Background**
- Algorithm Summary
- Results and Performance Analysis
- Conclusions

Background

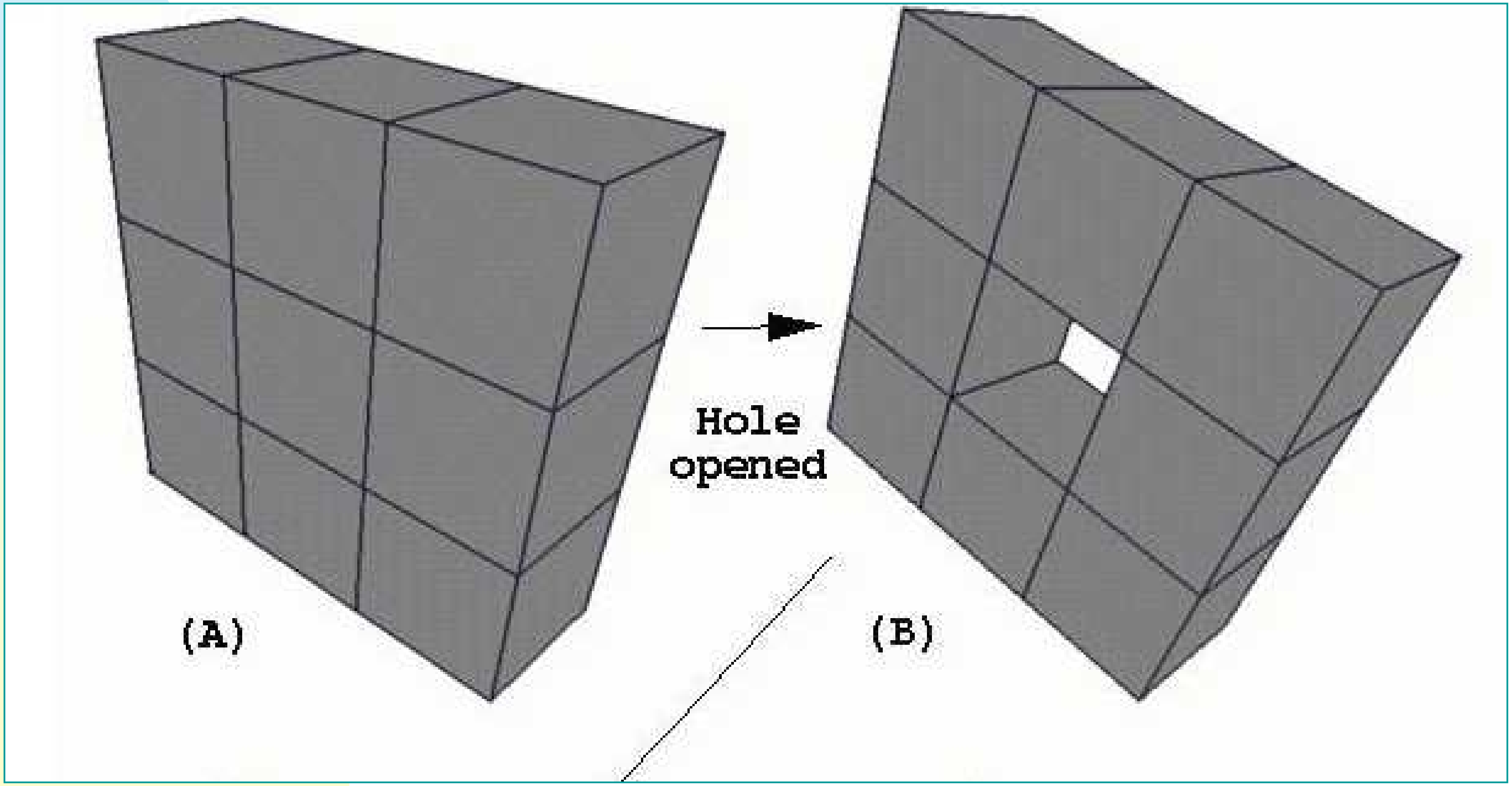
- Basic concepts
  - Manifold modeling
  - Non-manifold modeling
- Prior simplification algorithms
  - Vertex decimation
  - Vertex clustering
  - Iterative edge contraction

Manifold modeling

- Basic two-manifold model is widely used, it should follow three rules:
  - Planes only intersect at the vertices or edges.
  - Whole geometry is surrounded by planes, no losing or non-connected components
  - Every edge has exactly 2 faces (any number except 2 is not allowed).

Example of manifold model

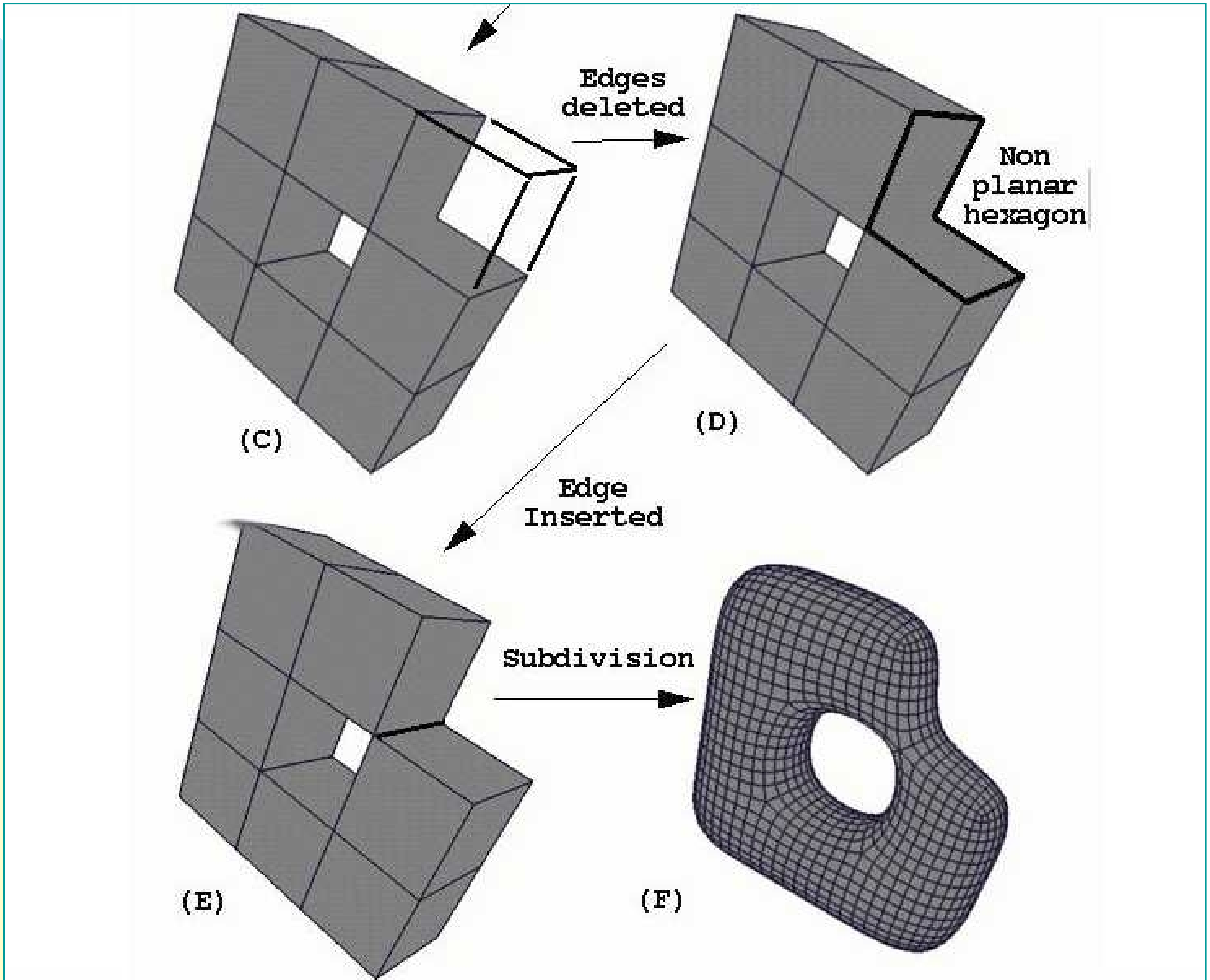
- A simple manifold model:



Non-manifold modeling

- Any model doesn't follow the previous 3 rules is an non-manifold model.
- Advantages:
  - Topology structure more flexible.
  - Allow infinite-thin space exists.
  - Open (non-connected) objects are allowed in the same structure.

Example of Non-manifold model



Simplification Algorithms (1)

- **Vertex Decimation**
  - Iteratively selects a vertex for removal, removes all adjacent faces, and re-triangulates the resulting hole.
  - Only works for manifold models since it carefully maintain the topology structure.

Simplification Algorithms (2)

- **Vertex Clustering**
  - a bounding box is placing around the original model and divided into grids. In each cell, clustering all vertices to a single vertex, then updates the model.
  - Fast, but the quality of output model is often low.
  - Doesn't work for non-manifold model.

Simplification Algorithms (3)

- **Iterative Edge Contraction**
  - Simplify the models by iteratively contracting selected edges. For example, progressive mesh algorithm, etc.

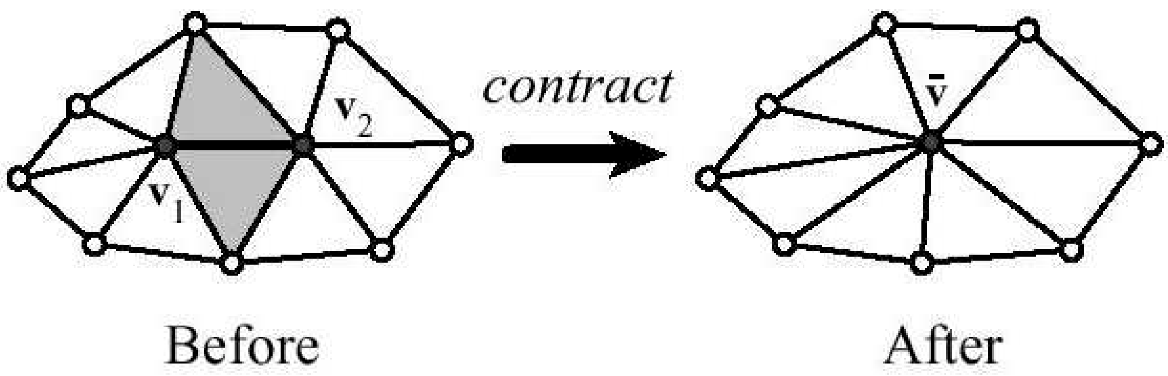


Figure 1: **Edge contraction.** The highlighted edge is contracted into a single point. The shaded triangles become degenerate and are removed during the contraction.

Outline

- Introduction
- Motivation
- Background
- **Algorithm Summary**
- Results and Performance Analysis
- Conclusions

Algorithm Summary

Assumptions about input model:

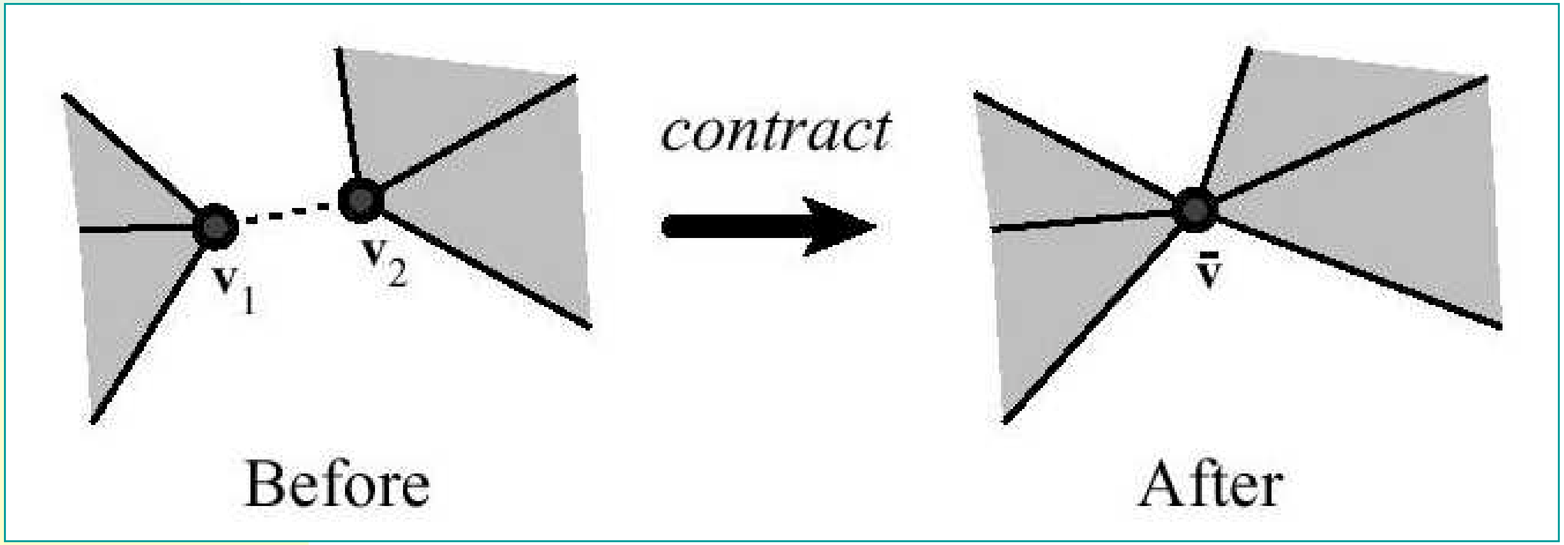
- Triangulated with valid connectivity.
- Surfaces need not have manifold topology. Therefore, Edges can border any number of faces.
- Vertices shared by arbitrary number of faces.

Basic Operations

- In order to deal with non-manifold models, authors introduced a process called **aggregation** to join unconnected regions of the model together.
- Aggregation is done by **Pair contraction**, could be edge contraction, or non-edge contraction.
- Use **Quadric Error Metrics (QEM)** as the criteria for pair contraction.

Pair Contraction

- Moves the vertices V1 and V2 to the new position V', connects all their incident edges to V1, and deletes the vertex V2.
- If (V1, V2) is an edge, use edge contraction. Else, use non-edge contraction.



Pair Selection

- A pair (V1, V2) is a valid pair for contraction if either:
  1. (V1, V2) is an edge, or
  2. $||V1-V2|| < t$, where t is a threshold parameter set by users.
- When t=0 means a simple edge contraction algorithm. Higher t value allow non-connected vertices to be paired.
- After this step we have a set of pairs.

How to measure errors?

Measure errors at current vertices.

For a given point v, measure sum of squared distances to associated set of planes.

- Each vertex v has an associated set of planes
  - initialize with planes of incident faces in original
  - merge sets when contracting pairs
  - initial error of each vertex is 0

How to measure errors?

For a given point v, measure sum of squared distances to associated set of planes.

- Locally, distance to plane equals distance to face

Measuring Error with Quadrics

Sum of squared distance to a set of planes

- Vertex v has associated set of planes

$$\Delta(\mathbf{v}) = \Delta([v_x \; v_y \; v_z \; 1]^{\mathsf{T}}) = \sum_{\mathbf{p} \in \text{planes}(\mathbf{v})} (\mathbf{p}^{\mathsf{T}}\mathbf{v})^2$$

- where p=[a,b,c,d] represents the plane defined by the equation ax+by+c+d=0

Measuring Error with Quadrics

$$\begin{aligned}\Delta(\mathbf{v}) &= \sum_{\mathbf{p}\in\text{planes}(\mathbf{v})} (\mathbf{v}^{\mathsf{T}}\mathbf{p})(\mathbf{p}^{\mathsf{T}}\mathbf{v}) \\ &= \sum_{\mathbf{p}\in\text{planes}(\mathbf{v})} \mathbf{v}^{\mathsf{T}}(\mathbf{p}\mathbf{p}^{\mathsf{T}})\mathbf{v} \\ &= \mathbf{v}^{\mathsf{T}}\left(\sum_{\mathbf{p}\in\text{planes}(\mathbf{v})} \mathbf{K}_{\mathbf{p}}\right)\mathbf{v}\end{aligned}$$

where $\mathbf{K}_{\mathbf{p}}$ is the matrix:

$$\mathbf{K}_{\mathbf{p}} = \mathbf{p}\mathbf{p}^{\mathsf{T}} = \begin{bmatrix} a^2 & ab & ac & ad \\ ab & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{bmatrix}$$

- Kp (fundamental error quadric) can be used to find the squared distance from any point in space to the plane p

What is a Quadric Error Metric?

- ***A quadric matrix Q is a 4x4 symmetric matrix assigns real number to every point v by:***

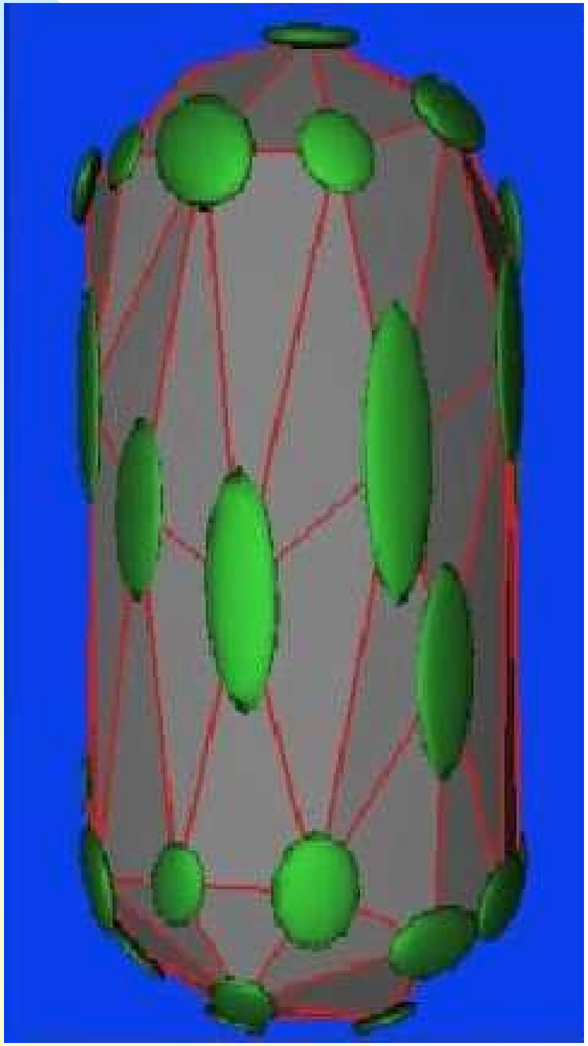
$$v^T Q v = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{12} & q_{22} & q_{23} & q_{24} \\ q_{13} & q_{23} & q_{33} & q_{34} \\ q_{14} & q_{24} & q_{34} & q_{44} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

What is a Quadric Error Metric?

- We get a second degree polynomial in x, y, and z
- The level surface is a quadric surface, it is defined as:

$$
\begin{aligned}
\mathbf{v}^T Q \mathbf{v} = {} & q_{11}x^2 + 2q_{12}xy + 2q_{13}xz + 2q_{14}x \\
& + q_{22}y^2 + 2q_{23}yz + 2q_{24} \\
& + q_{33}z^2 + 2q_{34}z + q_{44}
\end{aligned}
$$

What Are These Quadrics Really Doing?



- The ***level surface*** is almost always ellipsoids
  - when Q is positive definite
- Characterize error at vertex
  - vertex at center of each ellipsoid
  - move in anywhere on the ellipsoid with constant error

Algorithm Outline

Initialization

- Compute the Q matrices for all initial vertices
- Select all valid pairs (edges+non-edges)
- Computed minimal cost candidate for each pair (v1, v2). The cost of contracting a pair is defined as:

$$\boxed{\bar{\mathbf{v}}^{T}(\mathbf{Q}_1 + \mathbf{Q}_2)\bar{\mathbf{v}}}$$

Iteration

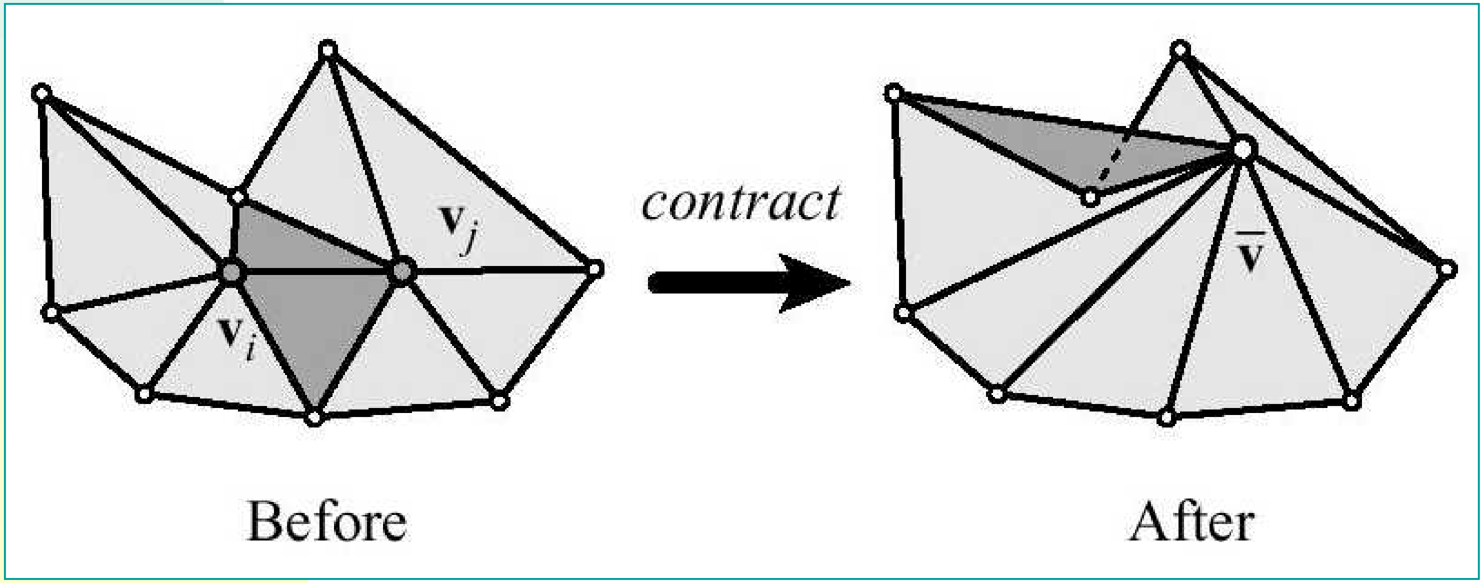
- select lowest cost pair (v1, v2)
- contract (v1, v2) -Q for new vertex is Q1+Q2
- Update all pairs involving v1&v2

Additional Algorithm Details

- **Open boundaries may be eaten away**
  - For each edge with a single incident face, assign large weight & add into endpoints

Additional Algorithm Details (2)

- **Contraction may fold mesh over on itself**
  - Check normal before & after, reject if any flip



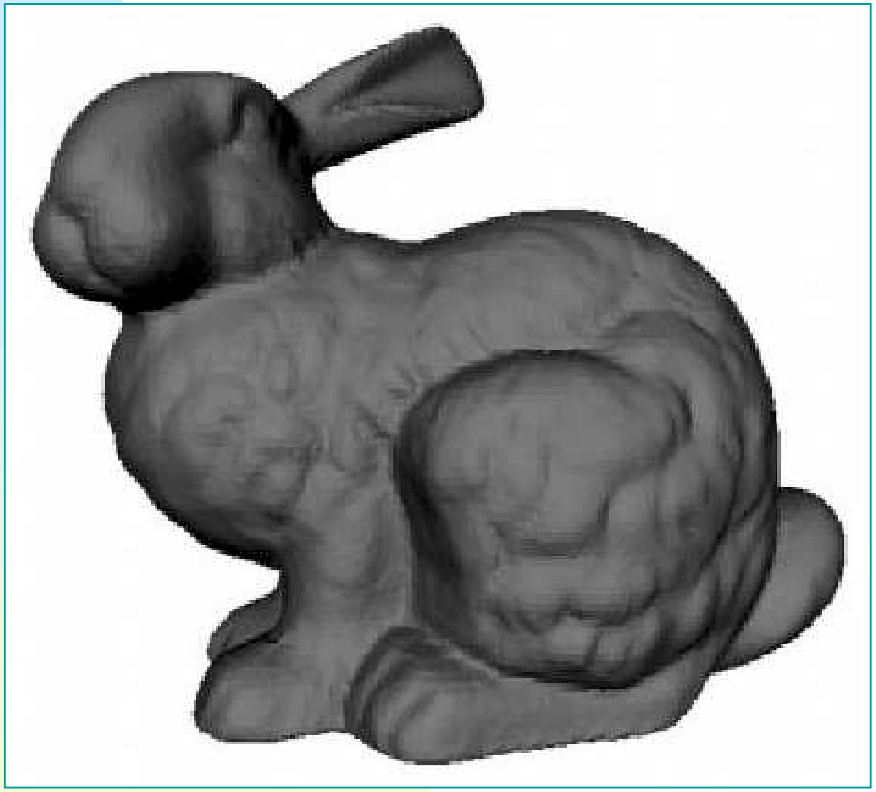
Outline

- Introduction
- Motivation
- Background
- Algorithm Summary
- **Results and Performance Analysis**
- Conclusions

Results and Performance Analysis

- Two approximation examples :
  - Stanford Bunny model
  - Human left foot model
- Handling surfaces with colored vertices
- The effect of pair thresholds
- Comparison with other algorithms

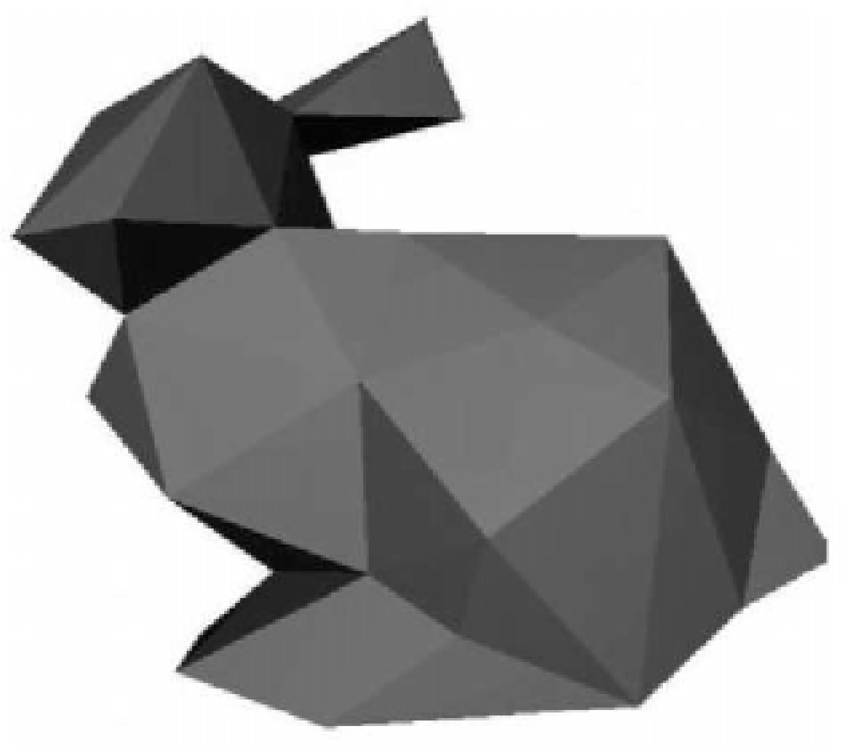
Stanford Bunny Model



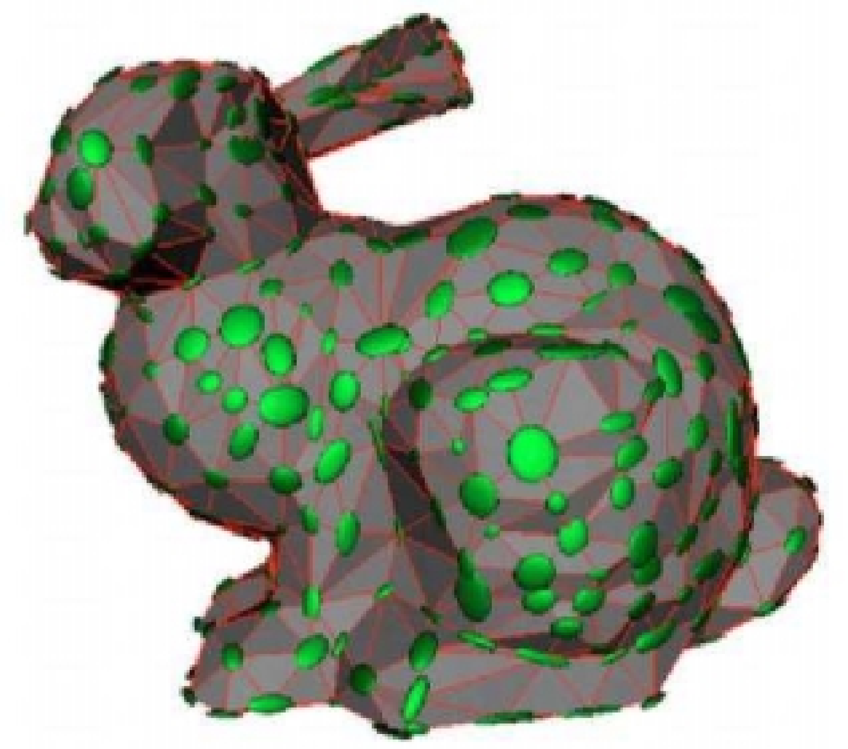
Original bunny model using 69,451 triangles.

An approximation using only 1000 triangles (1.4% of the original, generated in 15 seconds).

Stanford Bunny Model (2)

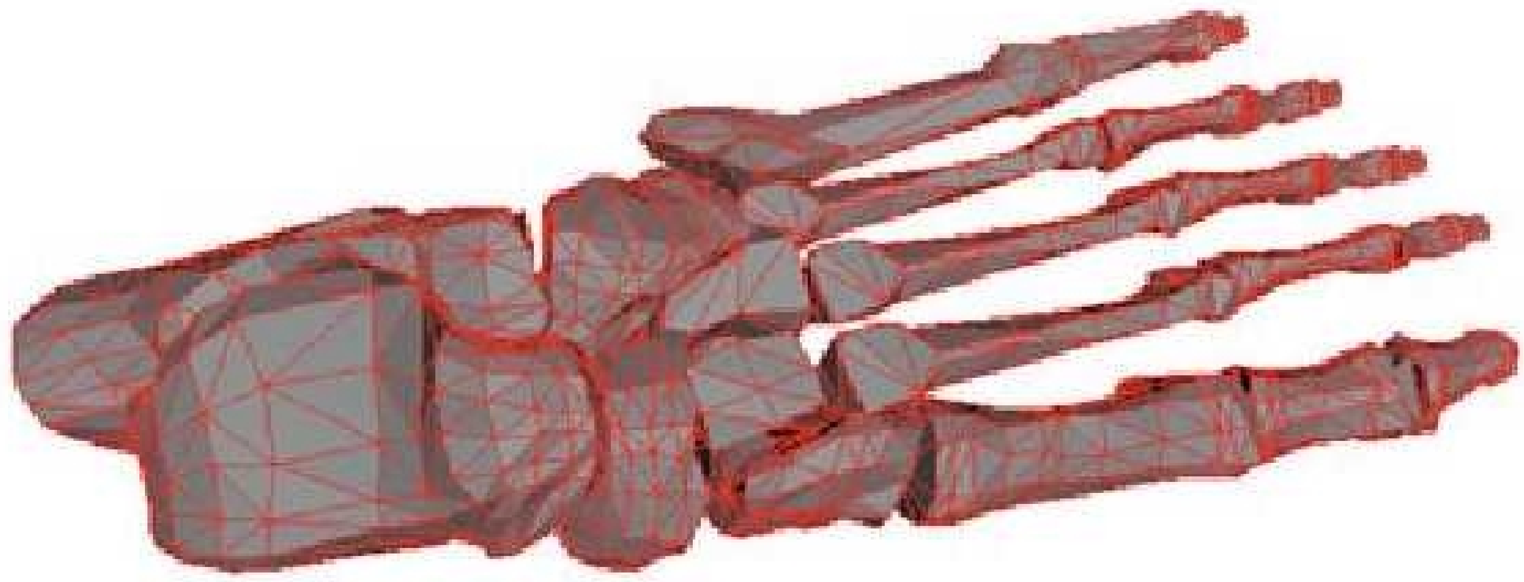


An approximation using only 100 triangles (0.14% of the original model, generated in 15 seconds).



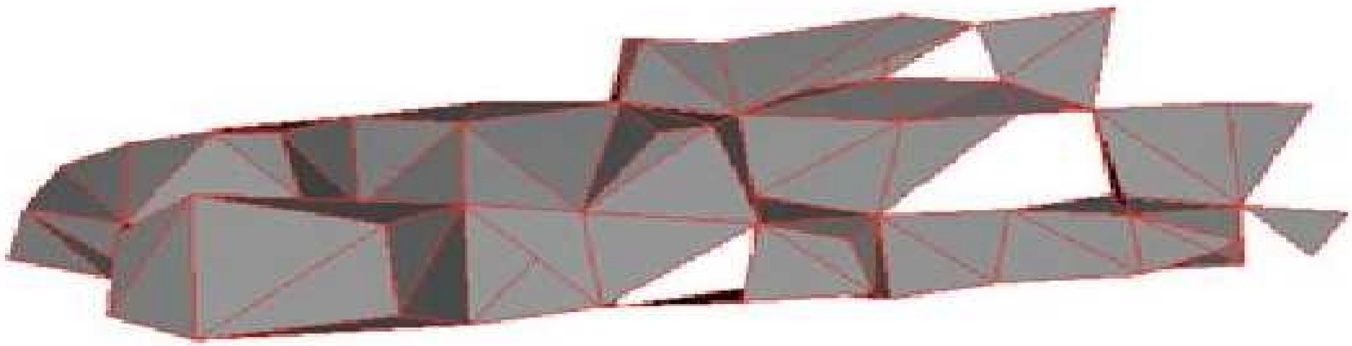
1 0 0 0 face approximation, error ellipsoids for each vertex are show in green.

Human's Left Foot Model

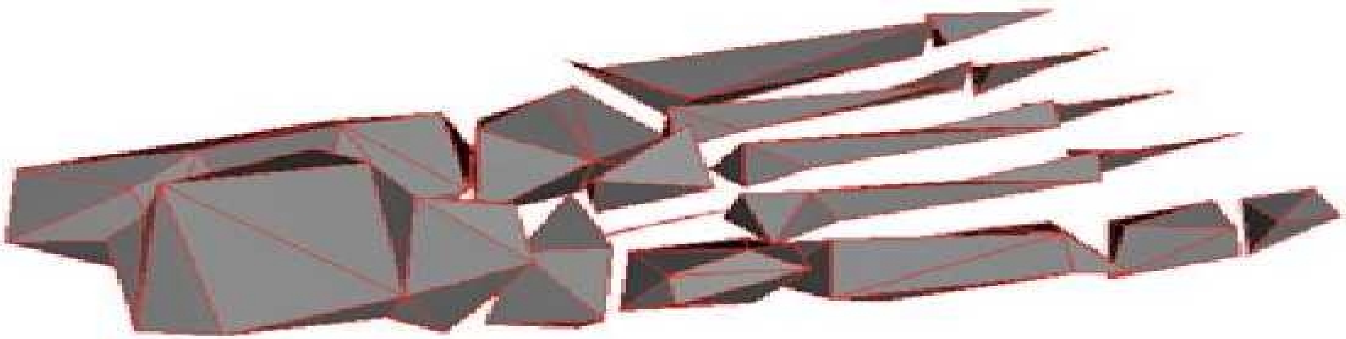


Original model. Bones of a human's left foot (4204 faces). Note that many separate bone segments.

Human’s Left Foot Model (2)

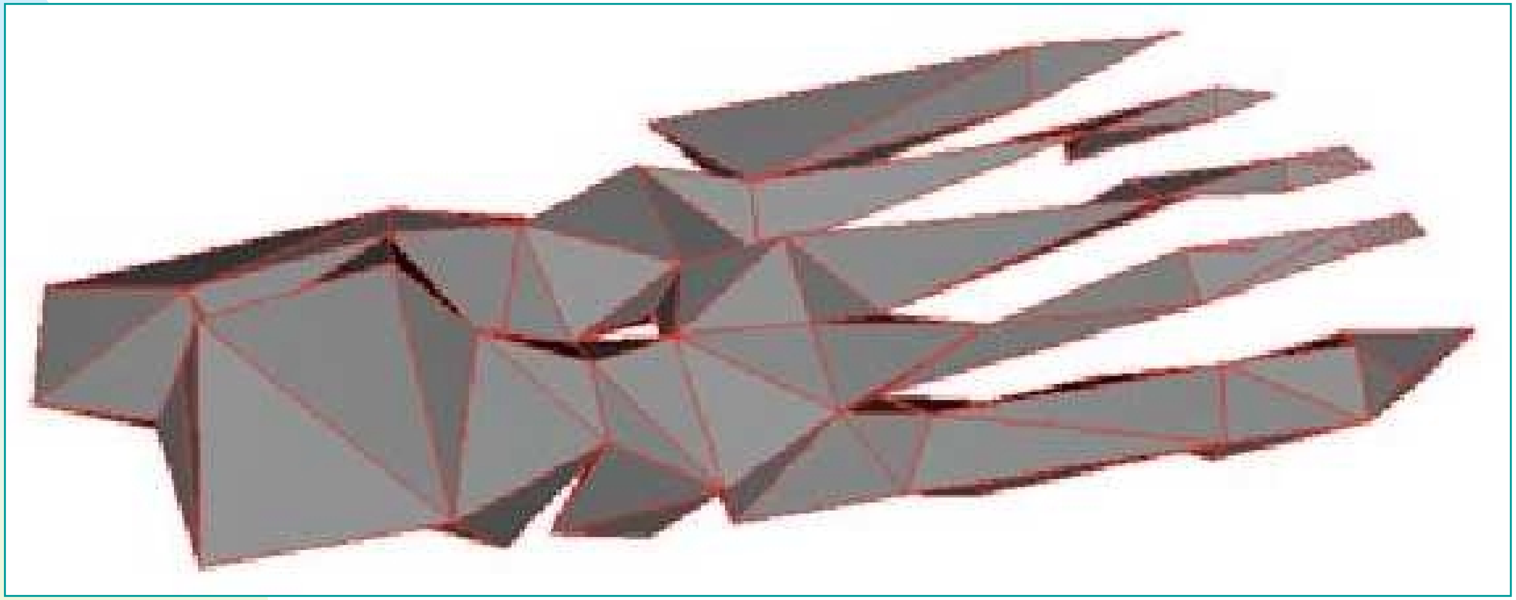


Uniform Vertex Clustering. 262 face approximation (11x4x4 grid). Indiscriminate joining destroys approximation quality.



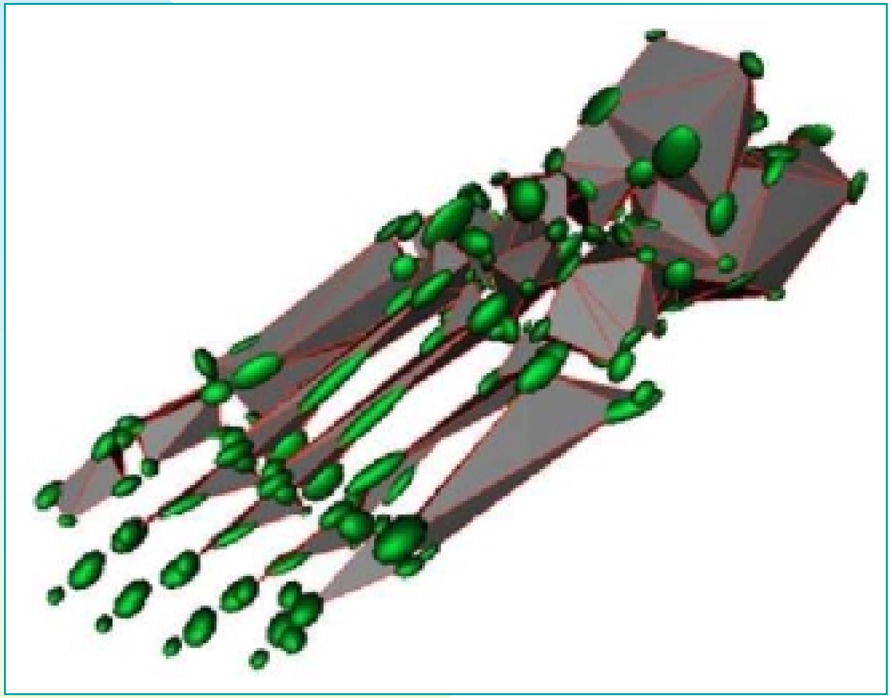
Edge Constructions. 250 face approximation. Bone segments at the ends of the toes have disappeared (receding). The hole opening occurred.

Human's Left Foot Model (3)

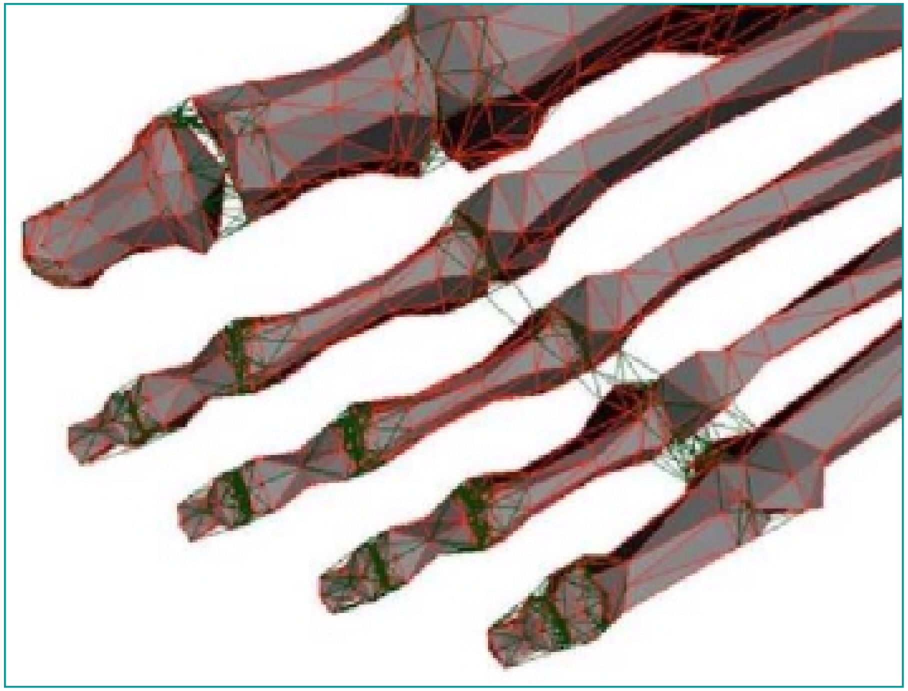


Pair Constructions. 250 face approximation (t=0.318). Toes are being merged into larger solid components. No receding and no hole opening. This model contains 61 non-manifold edges now.

Human's Left Foot Model (4)



Level surfaces of the error quadrics at the vertices of the approximation (250 triangles).



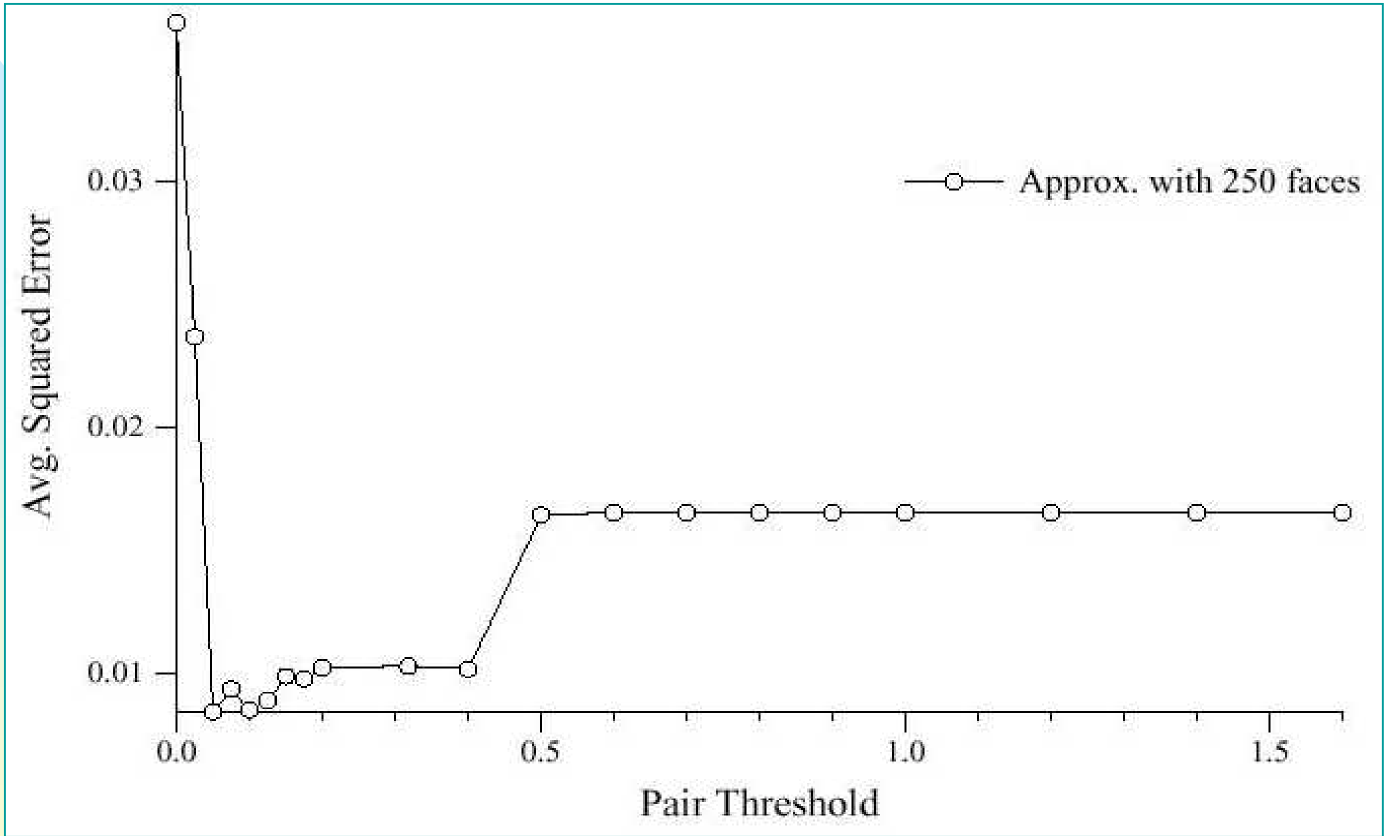
Pairs selected as valid during the initialization. Red pairs are edges, green pairs are non-edges.

Handling Surfaces with Colored Vertices

- Before: v=(x,y,z,1) and Q is a 4x4 matrix
- After: v=(x,y,z,r,g,b,1) and Q is a 7x7 matrix

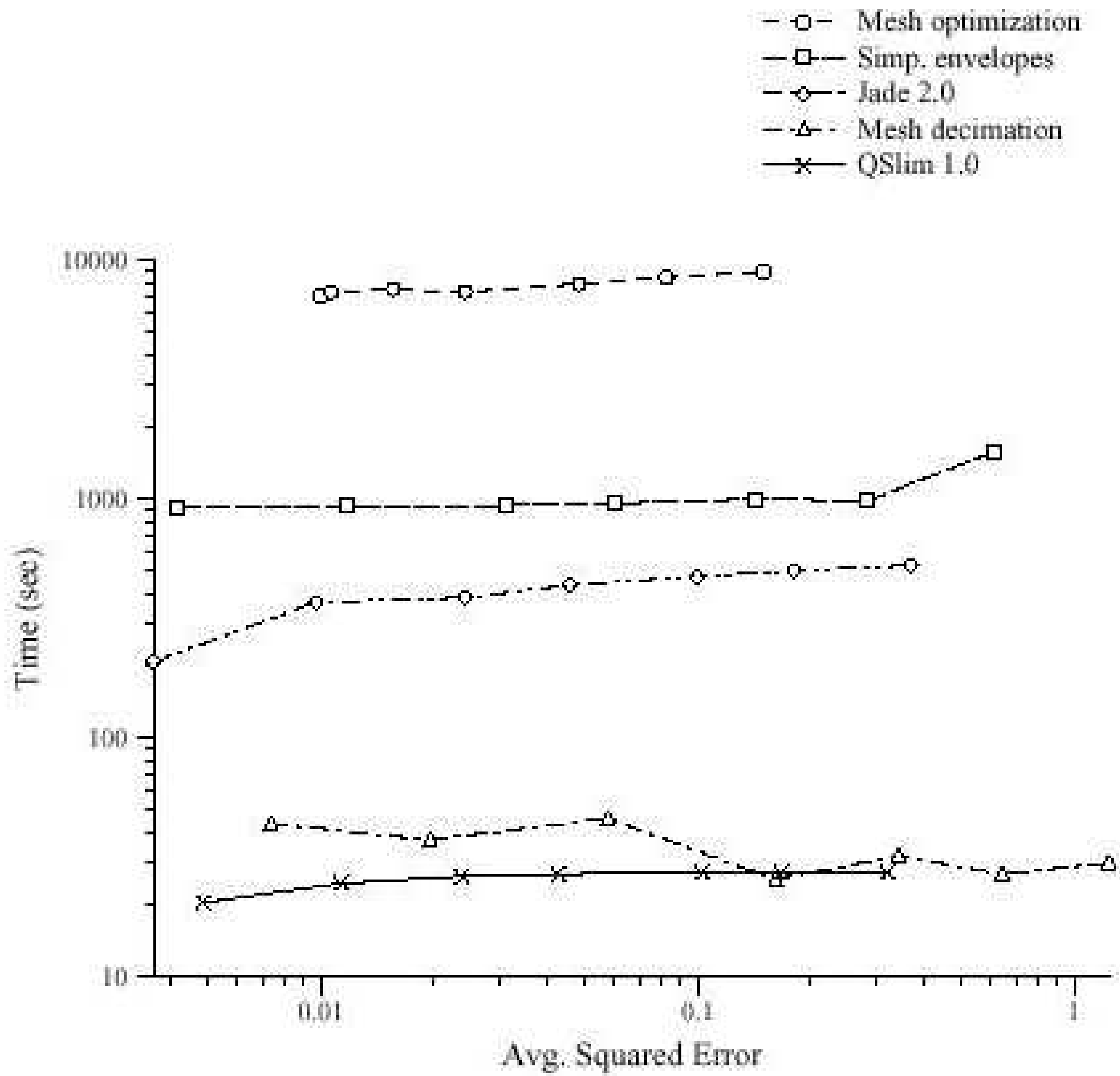
19,404 faces | 1,000 face approximation (5.2% of the original)

The Effect of Pair Thresholds



Approximation error E_{250} is shown as a function of t for a 250 face approximation of the foot model. Pair contractions resulting in aggregation can significantly reduce approximation error.

Comparison with Other Algorithms



Comparison with other methods: Tradeoff of approximation error vs. Running time for simplifying bunny model.

(This figure is taken from author’s PhD thesis *Quadric-Based Polygonal Surface Simplification* , 1998)

Conclusions

- Efficient algorithm with quality results
  - High quality approximations
  - Good compromise between highest quality and fastest simplification (provides a useful middle ground)
  - Simplifies both geometry and topology
  - Can also handle surface properties (such as color)
- Also has weaknesses:
  - measuring error as a distance to a set of planes only works well in a suitably local neighborhood
  - Input surface is rigid.

Reference

[1] Michael Garland and Paul S. Heckbert, *Surface Simplification Using Quadric Error Metrics*, Computer Graphics, volume 31, pages 209-216, 1997.

[2] Michael Garland, *Quadric-Based Polygonal Surface Simplification*, Phd Thesis, Carnegie Mellon University, 1998.

[3] http://www-viz.tamu.edu/faculty/ergun/research/topology/siggraph01/talk/5.html

[4] http://graphics.cs.uiuc.edu/~garland/papers/s97-quadric-talk.pdf

Questions