Lecture 20: Topology Continued, Mesh Data Structures

COMPSCI/MATH 290-04

Chris Tralie, Duke University

3/29/2016

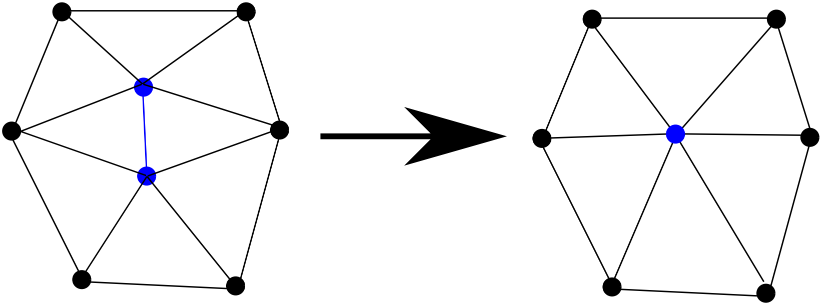
Announcements

- Group Assignment 2 Due Tomorrow 3/30
- First project milestone Friday 4/8/2016
- Merged Units 3+4 Into 1 (I'm traveling on 4/21)
- Attendance sheets

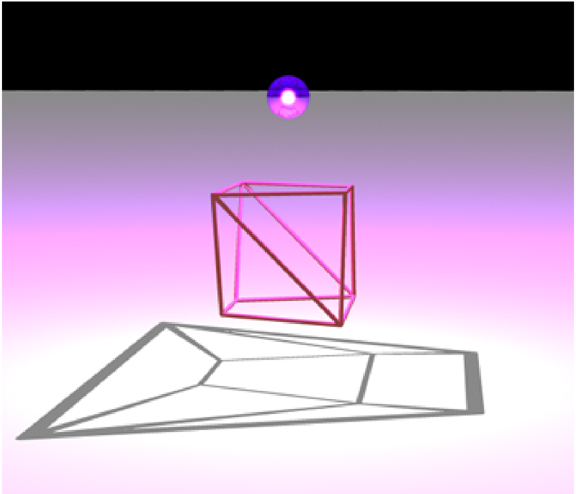
Table of Contents

Edge Collapse Case

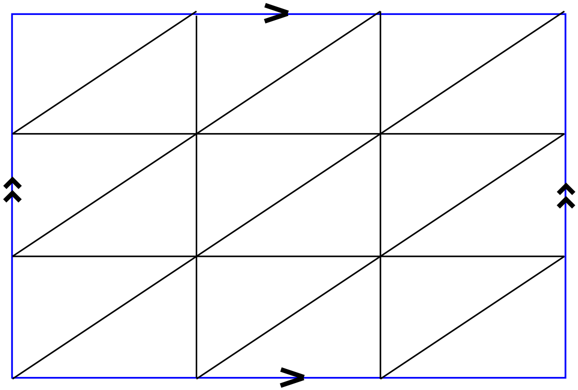
Planar graphs: $V - E + F = 2$



Review: Convex Shadow Casting (Stereographic Projection)

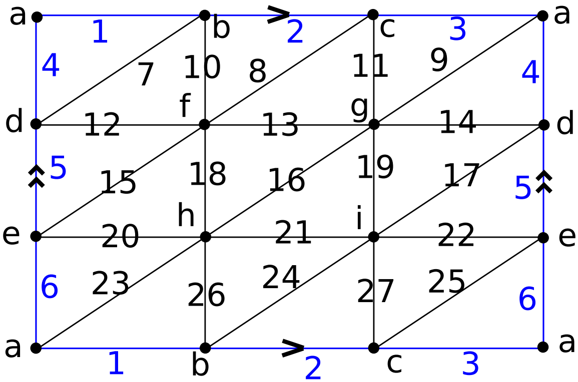


Review: Torus Euler Characteristic

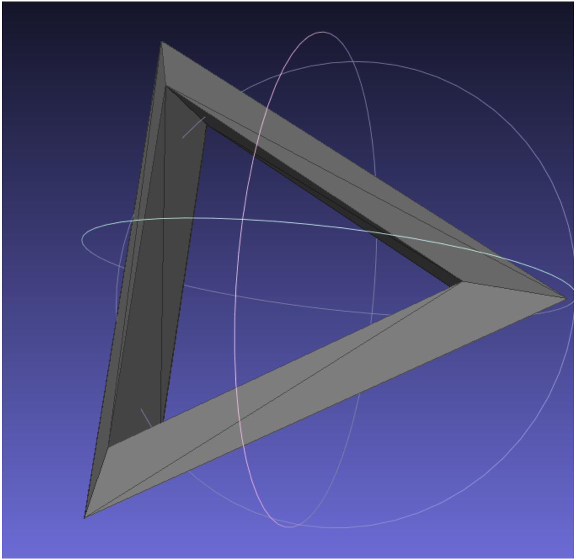


Review: Torus Euler Characteristic

9 vertices, 27 edges, 18 faces: $\chi = 0$

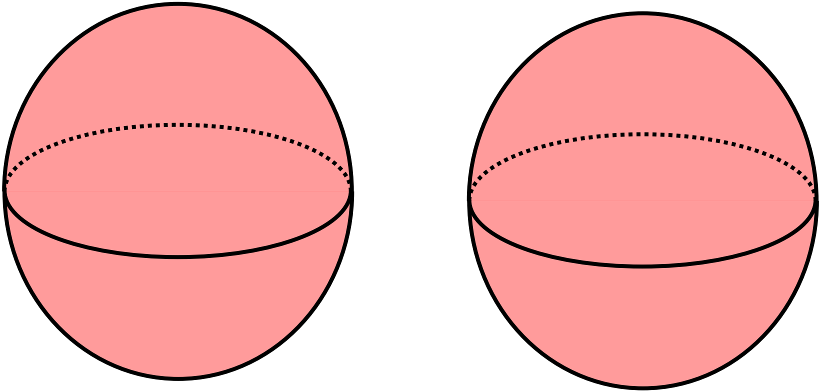


Review: Torus Euler Characteristic



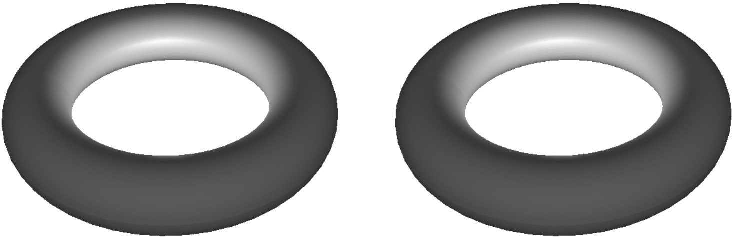
Duplicating Spheres

What's the euler characteristic of two spheres?



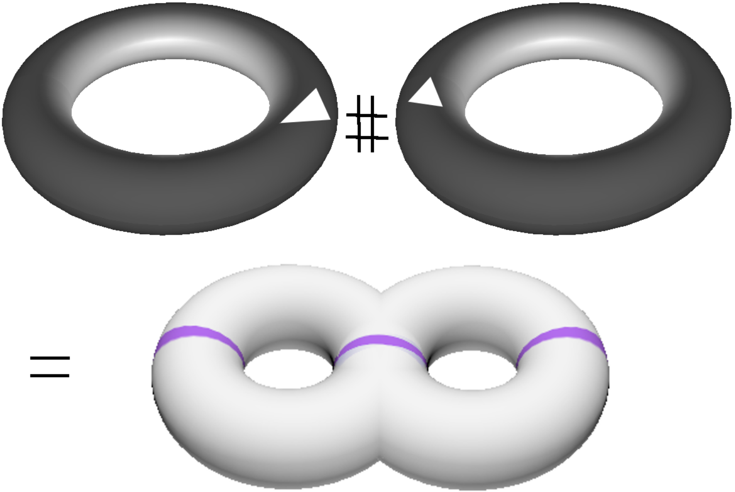
Duplicating Tori

What's the euler characteristic of two tori?



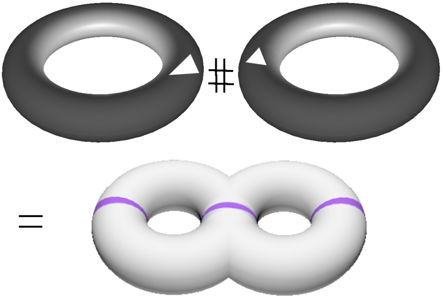
Connected Sum

$T_1 \# T_1 = T_2$



Connected Sum

$T_1 \# T_1 = T_2$
What is the Euler characteristic?



Connected Sum: g Tori

What is the Euler characteristic of $T_N = T_1 \# T_1 \# \ldots \# T_1$ g times?

Connected Sum: g Tori

What is the Euler characteristic of $T_N = T_1 \# T_1 \# \ldots \# T_1$ g times?

$$\chi = 2 - 2g$$

Connected Sum: g Tori

What is the Euler characteristic of $T_N = T_1 \# T_1 \# \ldots \# T_1$ g times?

$$\chi = 2 - 2g$$

- g is known as the "genus"

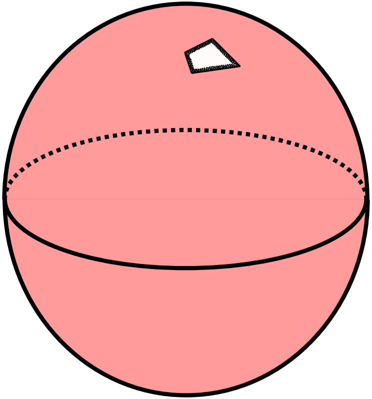
Connected Sum with Spheres

What is the connected sum of a sphere with a sphere?

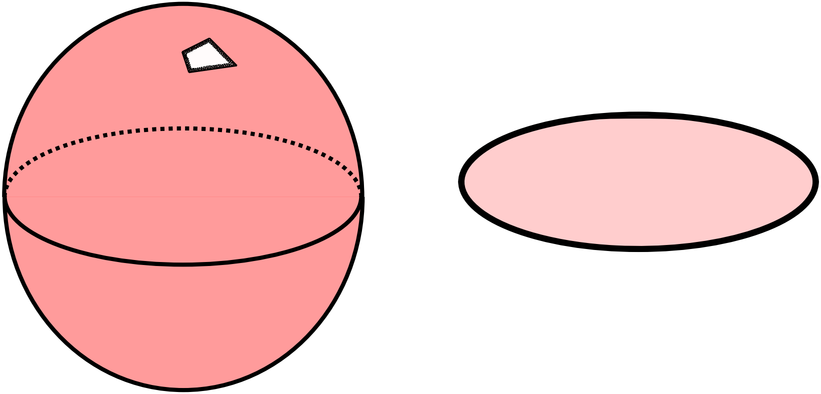
Connected Sum with Spheres

What is the connected sum of a torus with a sphere?

Boundaries / Discs



Boundaries / Discs



Euler Characteristic: Homology

$$\chi = \beta_0 - \beta_1 + \beta_2$$

- β_0: Number of connected components
- β_1: Number of independent loops/cycles
- β_2 Number of independent voids

Something With Euler Characteristic of 3?

Table of Contents

Order of Edges in Planar Graph

$$V - E + F = 2$$

Order of Edges in Planar Graph

$$V - E + F = 2$$

- There are at least 3 edges per face, and each edge is on the boundary of 2 faces for a *manifold mesh*

Order of Edges in Planar Graph

$$V - E + F = 2$$

- ▷ There are at least 3 edges per face, and each edge is on the boundary of 2 faces for a *manifold mesh*

$$3F \geq 2E \implies F \geq \frac{2}{3}E$$

Order of Edges in Planar Graph

$$V - E + F = 2$$

- There are at least 3 edges per face, and each edge is on the boundary of 2 faces for a *manifold mesh*

$3F \geq 2E \implies F \geq \frac{2}{3}E$
$V - E + \frac{2}{3}E \geq 2$

Order of Edges in Planar Graph

$$V - E + F = 2$$

- There are at least 3 edges per face, and each edge is on the boundary of 2 faces for a *manifold mesh*

$3F \geq 2E \implies F \geq \frac{2}{3}E$
$V - E + \frac{2}{3}E \geq 2$
$E \leq 3V - 6 \implies E = O(3V), F = O(2V)$

Vertices Per Polygon

Put all vertex coordinates for each polygon

x_{11}, y_{11}, z_{11}	x_{12}, y_{12}, z_{12}	x_{13}, y_{13}, z_{13}
x_{21}, y_{21}, z_{21}	x_{22}, y_{22}, z_{22}	x_{23}, y_{23}, z_{23}
$\ldots$	$\ldots$	$\ldots$
$\ldots$	$\ldots$	$\ldots$
x_{F1}, y_{F1}, z_{F1}	x_{F2}, y_{F2}, z_{F2}	x_{F3}, y_{F3}, z_{F3}

How many bytes per vertex, assuming 32-bit single precision floating point?

Vertices Per Polygon

Put all vertex coordinates for each polygon

x_{11}, y_{11}, z_{11}	x_{12}, y_{12}, z_{12}	x_{13}, y_{13}, z_{13}
x_{21}, y_{21}, z_{21}	x_{22}, y_{22}, z_{22}	x_{23}, y_{23}, z_{23}
$\dots$	$\dots$	$\dots$
$\dots$	$\dots$	$\dots$
x_{F1}, y_{F1}, z_{F1}	x_{F2}, y_{F2}, z_{F2}	x_{F3}, y_{F3}, z_{F3}

How many bytes per vertex, assuming 32-bit single precision floating point?

- ▷ 72 bytes/vertex

Basic “Off File” Index-Based Format

Vertices
$x_1\ y_1\ z_1$
...
$x_N\ y_N\ z_N$

Faces
$i_{11}\ i_{12}\ i_{13}$
$i_{21}\ i_{22}\ i_{23}$
...
...
...
$i_{F1}\ i_{F2}\ i_{F3}$

Basic “Off File” Index-Based Format

Vertices
$x_1\ y_1\ z_1$
...
$x_N\ y_N\ z_N$

Faces
$i_{11}\ i_{12}\ i_{13}$
$i_{21}\ i_{22}\ i_{23}$
...
...
...
$i_{F1}\ i_{F2}\ i_{F3}$

- ▷ 36 bytes/vertex

Basic “Off File” Index-Based Format

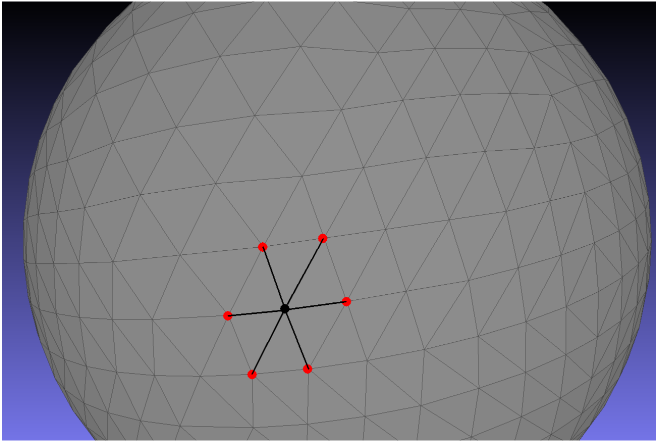
Vertices
x_1 y_1 z_1
...
x_N y_N z_N

Faces
i_{11} i_{12} i_{13}
i_{21} i_{22} i_{23}
...
...
...
i_{F1} i_{F2} i_{F3}

- ▷ 36 bytes/vertex
- ▷ Vertex buffers, index buffers in OpenGL

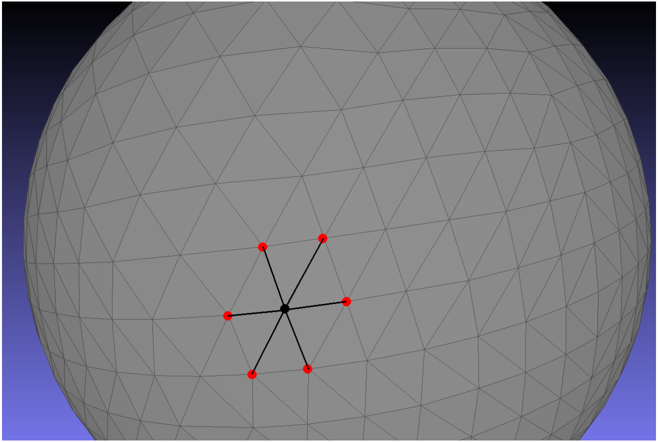
Query "One Ring Neighbors"

- A very common operation



Query "One Ring Neighbors"

- A very common operation

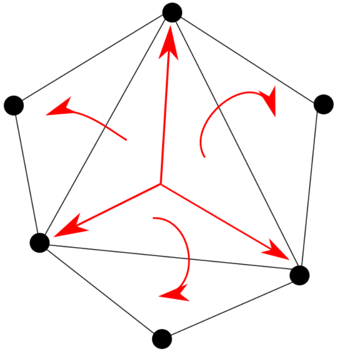


- Time complexity in vertex index scheme?

Face Adjacency

Vertex	
Point	position
FacePointer	face

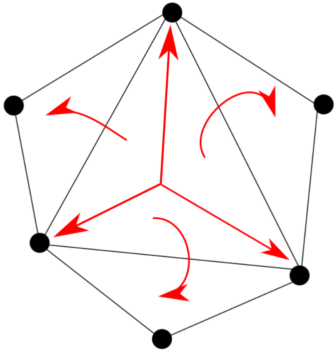
Face	
VertexPointer	vertices[3]
FacePointer	neighbors[3]



Face Adjacency

Vertex	
Point	position
FacePointer	face

Face	
VertexPointer	vertices[3]
FacePointer	neighbors[3]



- 24 bytes per face, 16 bytes per vertex = 64 bytes / vertex

GLEAT/S3DGLPY Format

Vertex	
Point	position
EdgePointer	edges[] (CCW)

Face	
VertexPointer	startVertex
EdgePointer	edges[] (CCW)

Edge	
VertexPointer	vertex1
VertexPointer	vertex2
FacePointer	face1
FacePointer	face2

GLEAT/S3DGLPY Format

Vertex	
Point	position
EdgePointer	edges[] (CCW)

Face	
VertexPointer	startVertex
EdgePointer	edges[] (CCW)

Edge	
VertexPointer	vertex1
VertexPointer	vertex2
FacePointer	face1
FacePointer	face2

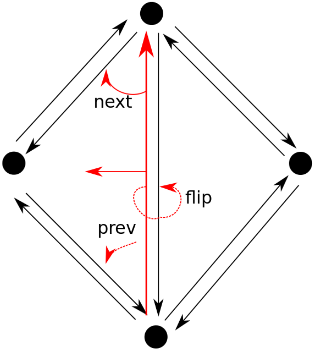
- 4*(3+6) bytes per vertex, 4*(1+3) bytes per face, 16 bytes per edge
- $36 + 16(2) + 16(3) = 116$ bytes/vertex

Half Edge Format

Vertex	
Point	position
HalfEdgePointer	halfedge

Face	
HalfedgePointer	halfedge

HalfEdge	
VertexPointer	vertex
FacePointer	face
HalfEdgePointer	next
HalfEdgePointer	previous
HalfEdgePointer	opposite

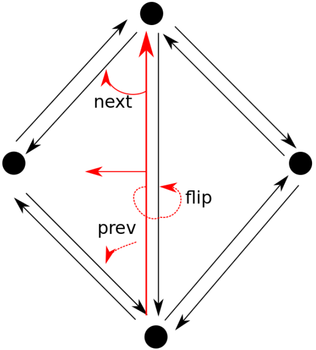


Half Edge Format

Vertex	
Point	position
HalfEdgePointer	halfedge

Face	
HalfedgePointer	halfedge

HalfEdge	
VertexPointer	vertex
FacePointer	face
HalfEdgePointer	next
HalfEdgePointer	previous
HalfEdgePointer	opposite



- 16 bytes per vertex, 4 bytes per face, 20 bytes per half-edge
- 16 + 4(2) + 20(3)(2 halfedges) = 76 bytes / vertex = 144 bytes/vertex

Half Edge One-Ring Neighbor