Johns Hopkins
Engineering for Professionals

605.767 Applied Computer Graphics

Brian Russin

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Module 7D
Subdivision Techniques

Normal, Texture, and Color Interpolation

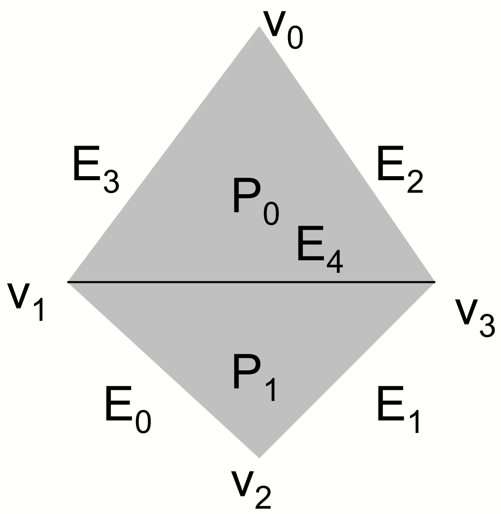
- Normals can be calculated explicitly for Loop and Modified Butterfly schemes
  - Involve expensive trig methods
    - Can improve efficiency by using a look-up table
- Another approach is to compute normals at the vertices of the control mesh
  - Use same subdivision method to subdivide normals as well
    - Need extra storage during subdivision
- Color and texture coordinates
  - Can also use subdivision method to create new color and texture coordinates
    - Treat color as a 4D vector: R,G,B,A
  - Linear interpolation may work best for Modified Butterfly scheme
    - Use an average of texture coordinates along edge where new vertex is created

Data Structures for Subdivision Surfaces

- Subdivision surfaces require information on neighboring vertices
    - Adjacency information
        - Several data structures allow efficient determination of vertices adjacent to one another
- Loop subdivision
    - Can use an adjacency structure
    - Faces store references to vertices and neighboring faces
    - Vertices store references to adjacent faces and vertices
- Other methods include Edge Tables and the Winged Edge data structure

Edge Tables

- Edge list
  - Points to the 2 vertices of an edge
  - Points to the 1 or 2 polygons adjacent to the edge
    - Exterior edge - one of the polygon pointers is NULL
- Vertex list
- Polygons
  - Point to edges that form the polygon



$V = (v_0, v_1, v_2, v_3)$

$E_0 = (v_1, v_2, P_1,\text{NULL})$

$E_1 = (v_2, v_3, P_1,\text{NULL})$

$E_2 = (v_0, v_3, P_0,\text{NULL})$

$E_3 = (v_0, v_1, P_0,\text{NULL})$

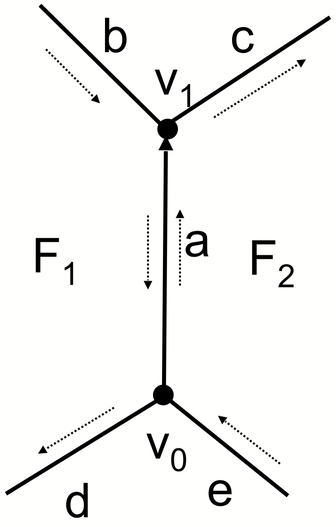
$E_4 = (v_1, v_3, P_0, P_1)$

$P_0 = (E_3, E_4, -E_2)$

$P_1 = (E_0, E_1, -E_4)$

Winged Edge Structure

- Complete topology encoded in the edges
  - Adjacency and connectivity fully described
- Each edge includes:
  - Pointers to vertices for the edge
  - Pointers to left and right faces
  - Pointers to the predecessor and successor edges when traversing its left face
  - Pointers to the predecessor and successor edges when traversing its right face
- Vertex table
  - Contains vertex position
  - Pointer to an incident edge
- Face table
  - Contains pointer to an edge on the boundary of the face
- Further info
  - http://www.cs.mtu.edu/~shene/COURSES/cs3621/NOTES/model/winged-e.html



Edge	**Vertices**		**Faces**		**Left Traverse**		**Right Traverse**	
Name	**Start**	**End**	**Left**	**Right**	**Pred**	**Succ**	**Pred**	**Succ**
A	v_0	v_1	F_1	F_2	b	d	e	c

Other Subdivision Topics

- Piecewise smooth subdivision
  - Change subdivision rules to create creases, corners, darts
    - Figure 17.50 (13.49 in 3rd Edition)
- Displaced Subdivision
  - Use displacement mapping to generate “bumpiness”
    - Displace vertices along normals
  - See Figure 17.52 (13.51 in 3rd Edition)
    - Loop subdivision is too smooth – displacement subdivision adds surface details
- Adaptive subdivision
  - Not all regions of a model need to be subdivided
  - Use some criteria and adaptively subdivide mesh where needed
    - Curvature
    - Screen size (make triangles < size of pixel)
    - View dependence
  - Must ensure that “cracks” aren’t produced

Applications of Subdivision Surfaces

- **From Intel white paper**
  - **Load Time Content Enhancement:**
    - Subdivision Surface technology can be used by a 3D software application to subdivide a base mesh either uniformly or adaptively to produce a mesh of a higher resolution. After the refined mesh has been generated, all computing and memory resources used in the process are freed. The refined mesh could then be rendered by the 3D application as it would any other mesh.
  - **Transmission and Compression:**
    - Subdivision Surface technology could be used as part of a strategy for compressing 3D models for Internet transmission. In this scenario, a server transmits the base mesh to a smart client. The client uses Subdivision Surface technology to "fill out" the base mesh by refining and smoothing it. A typical refinement may require three or four levels of uniform subdivision. Traditional mesh compression techniques may be applied to the base mesh for even greater mesh compression. The amount of subdivision can be set to the appropriate depth for the rendering performance of the client platform.

Applications of Subdivision Surfaces (cont.)

- **Enhancement of Existing 3D Content**
  - The primary requirement of the base mesh is that major surface features are represented by coarse polygons that provide a rough outline of the features. Many pre-existing models from real-time 3D applications already meet this criterion. Subdivision Surface technology can be used to add detail to these models, perhaps to make them more appropriate for rendering on higher performance processors.
- **Adaptive Subdivision of Static and Dynamic Meshes**
  - Static meshes are meshes whose vertex positions do not change between rendering frames. These include environment meshes such as terrain and objects on the terrain, as well as stationary buildings and their interiors. Dynamic meshes are meshes whose vertex positions are transformed dynamically and do not exhibit good or easily predicted frame-to-frame coherence. Examples include the skins of bones-based character animations and meshes modeling cloth. Adaptive subdivision works to dynamically generate a mesh that provides the same visual quality as a uniformly subdivided mesh, but with a smaller triangle count requiring fewer rendering sources. Adaptive subdivision adds surface detail only where needed based on a subdivision metric. Subdivision Surface technology provides some sample metrics, like screen space error, but developers are encouraged to write custom metrics using the simple metric plug-in interface.