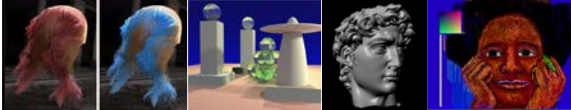


Advanced Computer Graphics

CSE 163 [Spring 2017], Lecture 10

Ravi Ramamoorthi

<http://www.cs.ucsd.edu/~ravr>



To Do

- Assignment 2 due May 19
 - Should already be well on way.
 - Contact us for difficulties etc.
 - This lecture is a "bonus": more advanced topic that is closer to the research frontier

Subdivision

- Was a very hot topic in computer graphics
- Brief survey lecture, quickly discuss ideas
- Detailed study quite sophisticated
 - See some of materials from readings

Advantages

- Simple (only need subdivision rule)
- Local (only look at nearby vertices)
- Arbitrary topology (since only local)
- No seams (unlike joining spline patches)

Video: Geri's Game ([outside link](#))

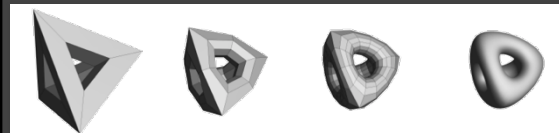


Outline

- *Basic Subdivision Schemes*
- Analysis of Continuity
- Exact and Efficient Evaluation (Stam 98)

Subdivision Surfaces

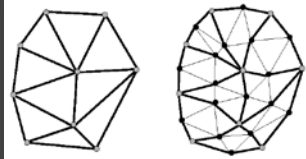
- Coarse mesh & subdivision rule
 - Smooth surface = limit of sequence of refinements



[Zorin & Schröder]

Key Questions

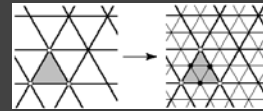
- How to refine mesh?
- Where to place new vertices?
 - Provable properties about limit surface



[Zorin & Schröder]

Loop Subdivision Scheme

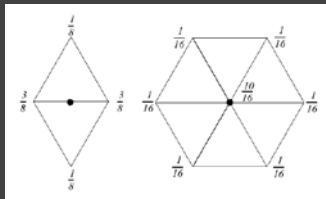
- How refine mesh?
 - Refine each triangle into 4 triangles by splitting each edge and connecting new vertices



[Zorin & Schröder]

Loop Subdivision Scheme

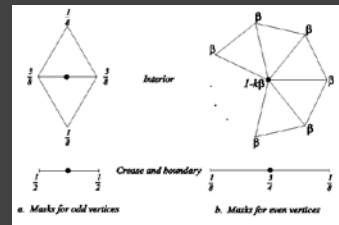
- Where to place new vertices?
 - Choose locations for new vertices as weighted average of original vertices in local neighborhood



[Zorin & Schröder]

Loop Subdivision Scheme

- Where to place new vertices?
 - Rules for *extraordinary vertices* and *boundaries*:



[Zorin & Schröder]

Loop Subdivision Scheme

Choose β by analyzing continuity of limit surface

- Original Loop

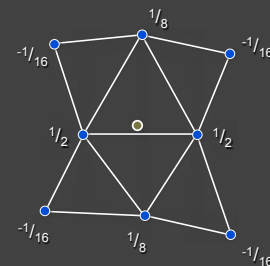
$$\beta = \frac{1}{n} \left(\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4} \cos \frac{2\pi}{n} \right)^2 \right)$$

- Warren

$$\beta = \begin{cases} \frac{3}{8n} & n > 3 \\ \frac{3}{16} & n = 3 \end{cases}$$

Butterfly Subdivision

- Interpolating subdivision: larger neighborhood



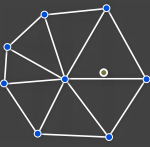
Modified Butterfly Subdivision

Need special weights near extraordinary vertices

- For $n = 3$, weights are $5/12, -1/12, -1/12$
- For $n = 4$, weights are $3/8, 0, -1/8, 0$
- For $n \geq 5$, weights are

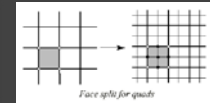
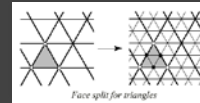
$$\frac{1}{n} \left(\frac{1}{4} + \cos \frac{2\pi j}{n} + \frac{1}{2} \cos \frac{4\pi j}{n} \right), j = 0 \dots n-1$$

- Weight of extraordinary vertex = $1 - \sum$ other weights



A Variety of Subdivision Schemes

- Triangles vs. Quads
- Interpolating vs. approximating



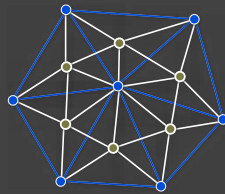
Face split		
	Triangular meshes	Quad. meshes
Approximating	Loop (C^2)	Catmull-Clark (C^2)
Interpolating	Mod. Butterfly (C^1)	Kobbelt (C^1)

Vertex split	
Doo-Sabin, Midedge (C^1)	Biquartic (C^2)

[Zorin & Schröder]

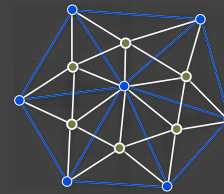
More Exotic Methods

- Kobbelt's subdivision:



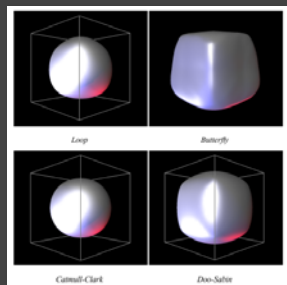
More Exotic Methods

- Kobbelt's subdivision:



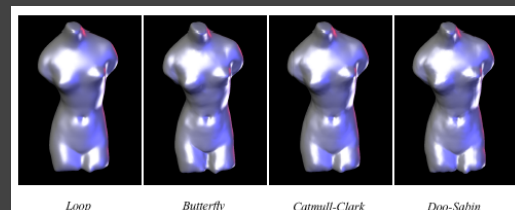
- Number of faces *triples* per iteration:
gives finer control over polygon count

Subdivision Schemes



[Zorin & Schröder]

Subdivision Schemes



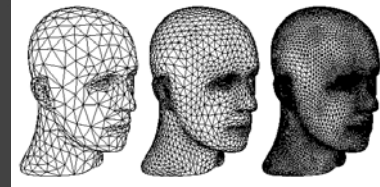
[Zorin & Schröder]

Outline

- Basic Subdivision Schemes
- Analysis of Continuity*
- Exact and Efficient Evaluation (Stam 98)

Analyzing Subdivision Schemes

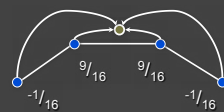
- Limit surface has provable smoothness properties



[Zorin & Schröder]

Analyzing Subdivision Schemes

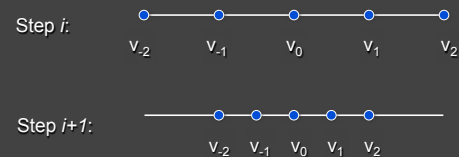
- Start with curves: 4-point interpolating scheme



(old points left where they are)

4-Point Scheme

- What is the support?



So, 5 new points depend on 5 old points

Subdivision Matrix

- How are vertices in neighborhood refined?
(with vertex renumbering like in last slide)

$$\begin{pmatrix} v_{-2}^{(i+1)} \\ v_{-1}^{(i+1)} \\ v_0^{(i+1)} \\ v_1^{(i+1)} \\ v_2^{(i+1)} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} v_{-2}^{(i)} \\ v_{-1}^{(i)} \\ v_0^{(i)} \\ v_1^{(i)} \\ v_2^{(i)} \end{pmatrix}$$

Subdivision Matrix

- How are vertices in neighborhood refined?
(with vertex renumbering like in last slide)

$$\tilde{\mathbf{v}}^{(i+1)} = \mathbf{S} \tilde{\mathbf{v}}^{(i)}$$

$$\text{After } n \text{ rounds: } \tilde{\mathbf{v}}^{(n)} = \mathbf{S}^n \tilde{\mathbf{v}}^{(0)}$$

Convergence Criterion

$$\vec{V}^{(n)} = \mathbf{S}^n \vec{V}^{(0)}$$

Expand in eigenvectors of $\mathbf{S}$:

$$\mathbf{S} = \sum_{i=0}^4 \lambda_i \mathbf{e}_i \mathbf{e}_i^T$$

$$\vec{V}^{(0)} = \sum_{i=0}^4 a_i \mathbf{e}_i$$

$$\vec{V}^{(n)} = \sum_{i=0}^4 a_i \lambda_i^n \mathbf{e}_i$$

$$\text{Criterion I: } |\lambda_i| \leq 1$$

Convergence Criterion

- What if all eigenvalues of $\mathbf{S}$ are < 1 ?
 - All points converge to 0 with repeated subdivision

$$\text{Criterion II: } \lambda_0 = 1$$

Translation Invariance

- For any translation t , want:

$$\begin{pmatrix} V_{-2}^{(i+1)} + t \\ V_{-1}^{(i+1)} + t \\ V_0^{(i+1)} + t \\ V_1^{(i+1)} + t \\ V_2^{(i+1)} + t \end{pmatrix} = \mathbf{S} \begin{pmatrix} V_{-2}^{(i)} + t \\ V_{-1}^{(i)} + t \\ V_0^{(i)} + t \\ V_1^{(i)} + t \\ V_2^{(i)} + t \end{pmatrix}$$

$$\vec{V}^{(i+1)} + t\vec{1} = \mathbf{S}(\vec{V}^{(i)} + t\vec{1})$$

$$\mathbf{S}\vec{1} = \vec{1}$$

$$\text{Criterion III: } \mathbf{e}_0 = \vec{1}, \text{ all other } |\lambda_i| < 1$$

Smoothness Criterion

$$\text{Plug back in: } \vec{V}^{(n)} = a_0 \mathbf{e}_0 + \sum_{i=1}^4 a_i \lambda_i^n \mathbf{e}_i$$

- Dominated by largest λ_i

- Case 1: $|\lambda_1| > |\lambda_2|$

$$\vec{V}^{(n)} = a_0 \mathbf{e}_0 + a_1 \lambda_1^n \mathbf{e}_1 + (\text{small})$$

- Group of 5 points gets shorter
- All points approach multiples of $\mathbf{e}_1 \rightarrow$ on a straight line
- Smooth!

Smoothness Criterion

- Case 2: $|\lambda_1| = |\lambda_2|$
 - Points can be anywhere in space spanned by $\mathbf{e}_1, \mathbf{e}_2$
 - No longer have smoothness guarantee

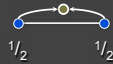
$$\text{Criterion IV: Smooth iff } \lambda_0 = 1 > |\lambda_1| > |\lambda_2|$$

Continuity and Smoothness

- So, what about 4-point scheme?
 - Eigenvalues = $1, 1/2, 1/4, 1/4, 1/8$
 - $\mathbf{e}_0 = \vec{1}$
 - Stable ✓
 - Translation invariant ✓
 - Smooth ✓

2-Point Scheme

- In contrast, consider 2-point interpolating scheme



- Support = 3
- Subdivision matrix =
$$\begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

Continuity of 2-Point Scheme

- Eigenvalues = 1, $\frac{1}{2}$, $\frac{1}{2}$
- $\mathbf{e}_0 = 1$
- Stable ✓
- Translation invariant ✓
- Smooth ✗
 - Not smooth; in fact, this is piecewise linear

For Surfaces...

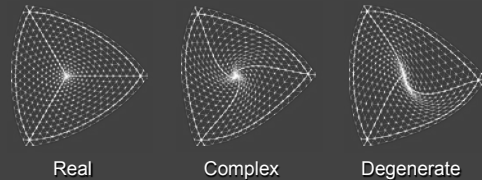
- Similar analysis: determine support, construct subdivision matrix, find eigenstuff
 - Caveat 1: separate analysis for each vertex valence
 - Caveat 2: consider more than 1 subdominant eigenvalue

Reif's smoothness condition: $\lambda_0 = 1 > |\lambda_1| \geq |\lambda_2| > |\lambda_j|$

- Points lie in subspace spanned by $\mathbf{e}_1$ and $\mathbf{e}_2$
 - If $|\lambda_1| \neq |\lambda_2|$, neighborhood stretched when subdivided, but remains 2-manifold

Fun with Subdivision Methods

Behavior of surfaces depends on eigenvalues



(recall that symmetric matrices have real eigenvalues)

[Zorin]

Outline

- Basic Subdivision Schemes
- Analysis of Continuity
- Exact and Efficient Evaluation (Stam 98)*

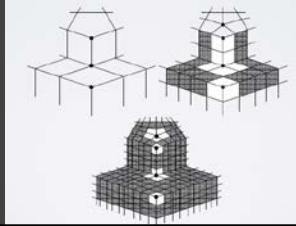
Slides courtesy James O'Brien

Practical Evaluation

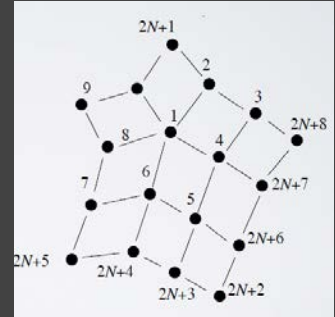
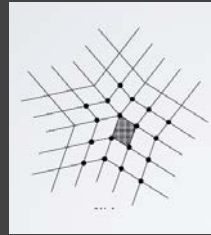
- Problems with Uniform Subdivision
 - Exponential growth of control mesh
 - Need several subdivisions before error is small
 - Ok if you are "drawing and forgetting", otherwise ...
- (Exact) Evaluation at arbitrary points
- Tangent and other derivative evaluation needed
- Paper by Jos Stam SIGGRAPH 98 efficient method
 - Exact evaluation (essentially take out "subdivision")
 - Smoothness *analysis* methods used to *evaluate*

Isolated Extraordinary Points

- After 2+ subdivisions, isolated “extraordinary” points where irregular valence
- Regular region is usually easy
 - For example, Catmull Clark can treat as B-Splines



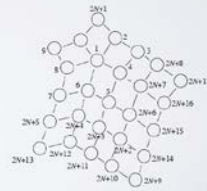
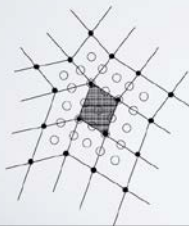
Isolated Extraordinary Points



Subdivision Matrix

$$C_1 = AC_0.$$

$$C_n = AC_{n-1} = A^n C_0$$



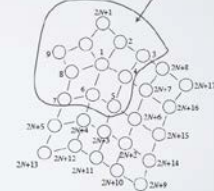
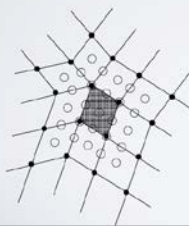
Subdivision Matrix

$$C_1 = AC_0.$$

$$C_n = AC_{n-1} = A^n C_0$$

$$A = \begin{pmatrix} S & 0 \\ S_{11} & S_{12} \end{pmatrix}$$

S is $2n \times 2n$



Eigen Space

$$C_1 = AC_0.$$

$$C_n = AC_{n-1} = A^n C_0$$

$$A = \begin{pmatrix} S & 0 \\ S_{11} & S_{12} \end{pmatrix}$$

$$AV = VA \rightarrow A = V\Lambda V^{-1}$$

$$\bar{C}_n = \bar{A}A^{n-1}C_0 = \bar{A}V\Lambda^{n-1}V^{-1}C_0$$

$$s_{k,n}(u, v) = \bar{C}_0^T \Lambda^{n-1} (P_k \bar{A} V)^T b(u, v)$$

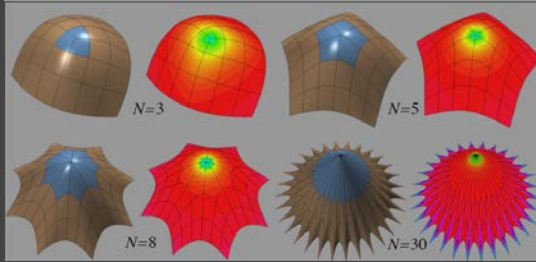
$$\bar{C}_0 = V^{-1}C_0$$

Only depends on valence of extraordinary vertex.

Comments

- Computing Eigen-Vectors is tricky
 - See Jos' paper for details
 - He includes solutions for valence up to 500
- All eigenvalues are (abs) less than one
 - Except for lead value which is exactly one
 - Well defined limit behavior
- Exact evaluation allows “pushing to limit surface”

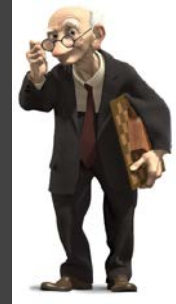
Curvature Plots



See Stam 98 for details

Summary

- Advantages:
 - Simple method for describing complex, smooth surfaces
 - Relatively easy to implement
 - Arbitrary topology
 - Local support
 - Guaranteed continuity
 - Multiresolution
- Difficulties:
 - Intuitive specification
 - Parameterization
 - Intersections



[Pixar]