CS155: Computer Graphics
Curves & Curved Surfaces
Z Sweedyk

Bezier Surfaces
Piece together smaller patches

Bezier Patch
Tensor product of Bezier curves

Bezier Patch problems
- Boundary conditions are difficult to maintain
- Variable resolution results in cracks
- Patches are necessarily quadrilaterals

Boundary conditions
$C & G^i (i>0)$ continuity require precise placement of control points

Boundary conditions
- Boundary conditions are difficult to maintain
- Solution: Enforce boundary constraints for user

This is essentially what B-splines do!
Subdivision Surfaces
Smooth surface is the limit of a subdivision process

Subdivision
User manipulates a coarse control mesh to affect the subdivided surface

Surface Subdivision
A smooth surface is computed by refining the control mesh using subdivision rules
1. Split face into multiple faces
2. Reposition vertices

Split Face
There are two approaches to splitting a face:
• Split vertices
• Split edges
Reposition vertices

- There are a variety of rules for repositioning vertices.
- This is tricky. Subdivision process must converge to a smooth surface!
- Rules typically “average” old vertex positions.
- Problem: it’s hard to create creases in subdivision surfaces.

Geri’s Game

see movie!

Loop algorithm

- Triangle mesh subdivision
- Edge splits
Loop Subdivision

Rule 1: Existing vertex
Case: Internal, valence 6

Note: Weights are applied to vertex positions in the old mesh.

4/20/2003 19

Loop Subdivision

Rule 1: Existing Vertex
Case: Internal, valence k

\[ K>3: \quad b=\frac{3}{8}k \]

\[ K=3: \quad b=\frac{3}{16} \]

4/20/2003 20

Loop Subdivision

Rule 1: Existing vertex
Case: Boundary

1/8 3/4 1/8

4/20/2003 21

Loop Subdivision

Rule 2: New vertex
Case: Internal

Note: Weights are applied to vertex positions in the old mesh.

4/20/2003 22

Loop Subdivision

Rule 2: New vertex
Case: Boundary

1/2 1/2

4/20/2003 23

Exercise 1

Use the Loop algorithm to split the faces of this triangle mesh.
Exercise 2

Your subdivided mesh should have a vertex corresponding to the one circled. Show how the position of this new vertex is calculated.

Exercise 3

Your subdivided mesh should have a vertex corresponding to the one circled. Show how the position of this new vertex is calculated.

Exercise 4

Your subdivided mesh should have a vertex corresponding to the edge indicated. Show how the position of this new vertex is calculated.

Exercise 5

Your subdivided mesh should have a vertex corresponding to the edge indicated. Show how the position of this new vertex is calculated.

Bezier: Boundary conditions

- Boundary conditions are difficult to maintain
- Solution: Enforce boundary constraints for user

This is essentially what B-splines do!

B-Splines

curve described by control points
**B-Splines**

these control points yield one curve segment

etc.

curves generally have $C^2$ continuity at the joins

- Uniform
- Non-uniform
Basis Matrix: $X_i(t)$

\[
\begin{bmatrix}
    a & b & c & d
\end{bmatrix} = \begin{bmatrix}
    1 & 3 & -3 & 1 \\
    3 & -6 & 3 & 0 \\
    -3 & 0 & 3 & 0 \\
    1 & 4 & 1 & 0
\end{bmatrix}
\]

\[X_i(t) = \frac{1}{6} \begin{bmatrix}
    1 & 3 & -3 & 1 \\
    3 & -6 & 3 & 0 \\
    -3 & 0 & 3 & 0 \\
    1 & 4 & 1 & 0
\end{bmatrix}
\]

Basis Functions: $X_i(t)$

\[X(t) = [t^3 \ t^2 \ t \ 1]\]

Basis Functions

- $B_0(t) = \frac{-t^3+3t^2-3t+1}{6} = \frac{(1-t)^3}{6}$
- $B_1(t) = \frac{3t^3-6t^2+4}{6}$
- $B_2(t) = \frac{-3t^3+3t^2+3t+1}{6}$
- $B_3(t) = \frac{u^3}{6}$

Blending Functions: $X_i(t)$

- One blending function
- Translates of blending function
- Restricted to interval
Basis Functions

\[ B_0 \quad B_1 \quad B_2 \quad B_3 \]

\[ t=0 \quad t=1 \]

Local Parameterization

\[ t \in [0,1] \]

Global Parameterization

\[ t \in [i, i+1] \quad t \in [i+1, i+2] \]

Uniform B-Spline

Knots: \([0, 1, 2, ..., n]\)

Parametric intervals of the curve segments are \([0,1], [1,2], ..., [n-1,n]\)

non-Uniform B-Spline

Knots: \([t_0, t_1, ..., t_n]\), \(t_0 \leq t_1 \leq ... \leq t_n\)

Parametric intervals of the curve segments are \([t_0, t_1], [t_1, t_2], ..., [t_{n-1}, t_n]\)

next time