

Lecture 15

Curves 3 – Polynomial Derivations, Bezier details, and B-Splines

Michael Gleicher
 October 27, 2005
 Notes prepared as notes: Not projected for class

Polynomial Segments

- Canonical form: $\sum a_i u^i$
 - General – but not convenient for control
- Blending function form: $\sum b_i(u) p_i$
 - Canonical functions are a special case u^0, u^1, \dots
 - Blending functions give easier to use points
- Example: line segment
 - Center, offset: $a_0 + a_1 u$
 - Endpoints: $b_0(u) p_0 + b_1(u) p_1$

Matrix form

- Can write canonical form as $\mathbf{a} \cdot \mathbf{u}$ $\mathbf{u} = [u^0 \ u^1 \ u^2 \ \dots]$
- Given definitions of p , solve for \mathbf{a} in terms of \mathbf{p}
 - Line segment example

$$f(0) = p_0$$

$$f(1) = p_1$$

Plug in canonical equations

$$p_0 = a_0 0^0 + a_1 0^1$$

$$p_1 = a_0 1^0 + a_1 1^1$$

Matrix form

$$\mathbf{p} = \mathbf{C} \mathbf{a}$$

Where

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

Basis Matrices

- The matrix \mathbf{C} is called the constraint matrix
- The inverse of \mathbf{C} is called \mathbf{B} , the Basis Matrix
 - $\mathbf{a} = \mathbf{B} \mathbf{p}$
- Since $f(u) = \mathbf{u} \cdot \mathbf{a}$
 - $f(u) = \mathbf{u} \mathbf{B} \mathbf{p}$ $\mathbf{u} \mathbf{B}$ are the blending functions

- In the example, $\mathbf{C} = [1 \ 0 \ 0 \ 1]$ $\mathbf{B} = \mathbf{C}^{-1}$

$$-\mathbf{B} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}$$

$$b_0(u) = 1 - u$$

$$b_1(u) = u$$

More complicated example: Catmull Rom Splines

- $f(0) = p_1$
- $f(1) = p_2$
- $f'(0) = \frac{1}{2} (p_2 - p_0)$
- $f'(1) = \frac{1}{2} (p_3 - p_1)$

- Remember...
 $f(u) = a_0 + a_1 u + a_2 u^2 + a_3 u^3$
- So
 $f'(u) = a_1 + 2 a_2 u + 3 a_3 u^2$

- $p_0 = f(1) - 2 f'(0)$
- $p_3 = f(0) + 2 f'(1)$

1	1 - 2	1	1
1			
1	1	1	1
1	2 (1)	2(2)	2(3)

Catmull Rom Blending Function

- $\mathbf{B} = \mathbf{C}^{-1}$

0	1	0	0
-1/2	0	1/2	0
1	-2 1/2	2	-1/2
-1/2	1 1/2	-1 1/2	1/2

$$b_0(u) = -\frac{1}{2}u + u^2 - \frac{1}{2}u^3$$

$$b_1(u) = 1 - 2 \frac{1}{2} u^2 + 1 \frac{1}{2} u^3$$

$$b_2(u) = \frac{1}{2} u - 2 u^2 - 1 \frac{1}{2} u^3$$

$$b_3(u) = -\frac{1}{2} u^2 + \frac{1}{2} u^3$$

Can we do Beziers this way?



- Yes – set up constraints and solve
 - $f(0) = p_0$
 - $f(1) = p_3$
 - $f'(0) = 3(p_1 - p_0)$
 - $f'(1) = 3(p_3 - p_2)$
- Doesn't generalize well (OK for 2,3,4)
- General form for Bezier blending functions
 - Bernstein Basis Polynomials

$$b_{k,n}(u) = C(n,k) u^k (1-u)^{n-k}$$

$$C(n,k) = \frac{n!}{k! (n-k)!}$$

Natural Cubics



- Can get C(2) interpolating cubics – just not local
- Define each segment such that:
 - $p_0 = f(0)$
 - $p_1 = f(1)$
 - $p_2 = f'(0)$
 - $p_3 = f'(1)$
- Figure out beginning derivatives of piece n+1 from the end of piece n
- Changes propagate (change beginning, effects end)

How do we make Smooth Curves?

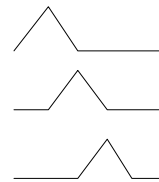


- Will be approximating
- Want flexibility
 - Any number of points
 - Any degree of continuity
- Want good properties
 - Locality
 - Convex Hull
 - Variation Diminishing
 - Shift Invariant (sum of blending functions = 1)

An example of blending functions



- Lines
- Blending functions are "bumps"
- Each piece is a spline
 - Two polynomials, degree 1
- Locality
 - Non-Zero over small range
 - Between 3 knots (d+2)
- Smoothness
 - D-1 continuity
- Shift invariant
- Convex Hull Property
- Shiftable (periodic)



B-Splines



- General scheme for generating blending functions
 - Any number of points (need more points than degree)
 - Any degree of polynomial (higher degree = smoother)
 - Any knot vector
- Blending function of degree D are B-Splines
 - Made of D+1 segments (span D+2 knots)
 - Each segment is a degree D polynomial
 - Only D+1 of them are non-zero at any time
 - Sum to one
 - Zero outside of the range
 - D-1 continuous
- Note: usually talk about "order" (degree+1)

Knot vectors



- Allow us to assign parameter values to points
- Makes it possible to alter the set of points but keep parameter values fixed
- Allows us to alter the spacing
- Allows us to create discontinuities
- (picture with lines)
- Uniform vs. Non-Uniform

Creating B-Splines

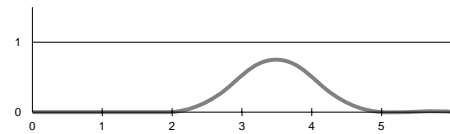


- Cox-de Boor recurrence
- Convolutions of the unit box
- When $d > 1$, the functions do not interpolate
 - They never reach the value of 1

Quadratic, Uniform B-Splines



$$b_{i,3}(t) = \begin{cases} \frac{1}{2}u^2 & \text{if } i \leq t < i+1 & u = t - i \\ -u^2 + u + .5 & \text{if } i+1 \leq t < i+2 & u = t - (i+1) \\ \frac{1}{2}(1-u)^2 & \text{if } i+2 \leq t < i+3 & u = t - (i+2) \\ 0 & \text{otherwise.} \end{cases}$$



Even More General?



- B-Splines cannot represent conic sections
 - Can't make an exact circle
- Express curves / surfaces as the RATIO
- Non-Uniform Rational B-Spline Surfaces
 - (NURBS)
- Extensions to surfaces later in the class

Using B-Splines



- Figure out closed form basis functions
 - Rather than using Cox-de Boor
- Can encode into a Basis matrix
 - But cannot derive the same way
- Periodic basis functions are nice
 - Implement once
- Gives a nice way to get very smooth curves
 - Cubics (usually) in graphics to provide C(2)