CS-184: Computer Graphics

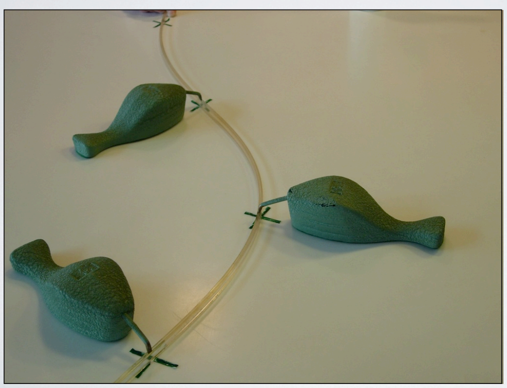
Lecture 13: Natural Splines, B-Splines, and NURBS

Maneesh Agrawala
University of California, Berkeley

Slides based on those of James O'Brien

Natural Splines

Draw a "smooth" line through several points



A real draftsman's spline.

Image from Carl de Boor's webpage.

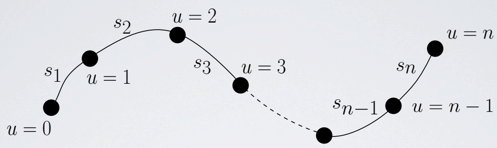
Natural Cubic Splines

Given $n+1$ points

- Generate a curve with n segments
- Curves passes through points
- Curve is C^2 continuous

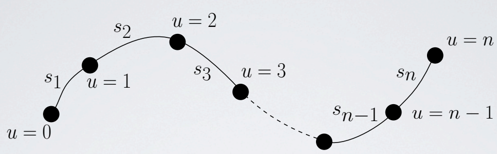
Use cubics because lower order is better...

Natural Cubic Splines



$$\mathbf{x}(u) = \begin{cases} \mathbf{s}_1(u) & \text{if } 0 \le u < 1 \\ \mathbf{s}_2(u-1) & \text{if } 1 \le u < 2 \\ \mathbf{s}_3(u-2) & \text{if } 2 \le u < 3 \\ & \vdots \\ \mathbf{s}_n(u-(n-1)) & \text{if } n-1 \le u \le n \end{cases}$$

Natural Cubic Splines



$s_i(0) = p_{i-1}$ $\quad i = 1 \ldots n$ $\quad \leftarrow$ n constraints

$s_i(1) = p_i$ $\quad i = 1 \ldots n$ $\quad \leftarrow$ n constraints

$s_i'(1) = s_{i+1}'(0)$ $\quad i = 1 \ldots n-1$ $\quad \leftarrow$ n-1 constraints

$s_i''(1) = s_{i+1}''(0)$ $\quad i = 1 \ldots n-1$ $\quad \leftarrow$ n-1 constraints

$s_1''(0) = s_n''(1) = 0$ $\quad \leftarrow$2 constraints

Total 4n constraints

Natural Cubic Splines

Interpolate data points

No convex hull property

Non-local support

- Consider matrix structure...

C^2 using cubic polynomials

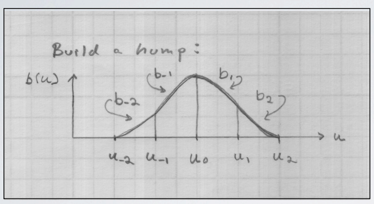
B-Splines

Goal: C^2 cubic curves with local support

- Give up interpolation
- Get convex hull property

Build basis by designing "hump" functions

B-Splines



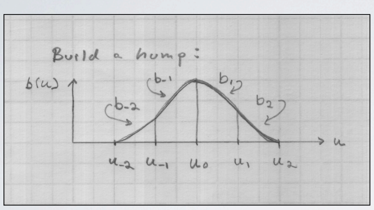
$$\mathbf{b}(u) = \begin{cases} \mathbf{b_{-2}}(u) & \text{if } u_{-2} \leq u < u_{-1} \\ \mathbf{b_{-1}}(u) & \text{if } u_{-1} \leq u < u_0 \\ \mathbf{b_{+1}}(u) & \text{if } u_0 \leq u < u_{+1} \\ \mathbf{b_{+2}}(u) & \text{if } u_{+1} \leq u \leq u_{+2} \end{cases}$$

$b''_{-2}(u_{-2}) = b'_{-2}(u_{-2}) = b_{-2}(u_{-2}) = 0$ $\leftarrow 3$ constraints
$b''_{+2}(u_{+2}) = b'_{+2}(u_{+2}) = b_{+2}(u_{+2}) = 0$ $\leftarrow 3$ constraints

$b_{-2}(u_{-1}) = b_{-1}(u_{-1})$
$b_{-1}(u_0) = b_{+1}(u_0)$ $\leftarrow$ [Repeat for b' and b''; 3×3=9 constraints
$b_{+1}(u_{+1}) = b_{+2}(u_{+1})$

Total 15 constraints ...... need one more

B-Splines



$$\mathbf{b}(u) = \begin{cases} \mathbf{b_{-2}}(u) & \text{if } u_{-2} \leq u < u_{-1} \\ \mathbf{b_{-1}}(u) & \text{if } u_{-1} \leq u < u_0 \\ \mathbf{b_{+1}}(u) & \text{if } u_0 \leq u < u_{+1} \\ \mathbf{b_{+2}}(u) & \text{if } u_{+1} \leq u \leq u_{+2} \end{cases}$$

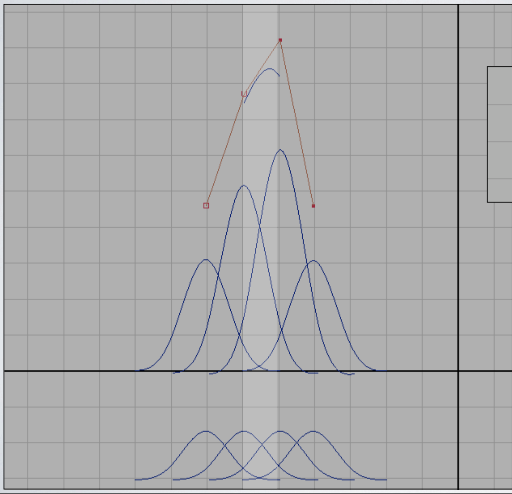
$b''_{-2}(u_{-2}) = b'_{-2}(u_{-2}) = b_{-2}(u_{-2}) = 0$ $\leftarrow 3$ constraints
$b''_{+2}(u_{+2}) = b'_{+2}(u_{+2}) = b_{+2}(u_{+2}) = 0$ $\leftarrow 3$ constraints

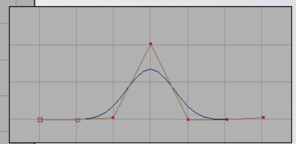
$b_{-2}(u_{-1}) = b_{-1}(u_{-1})$
$b_{-1}(u_0) = b_{+1}(u_0)$ $\leftarrow$ [Repeat for b' and b''; 3×3=9 constraints
$b_{+1}(u_{+1}) = b_{+2}(u_{+1})$

$b_{-2}(u_{-2}) + b_{-1}(u_{-1}) + b_{+1}(u_0) + b_{+2}(u_{+1}) = 1$ $\leftarrow 1$ constraint (convex hull)

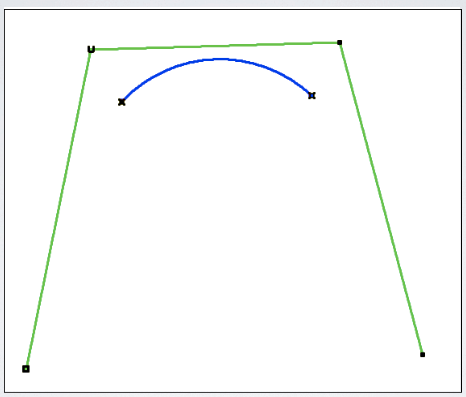
Total 16 constraints

B-Spline bases

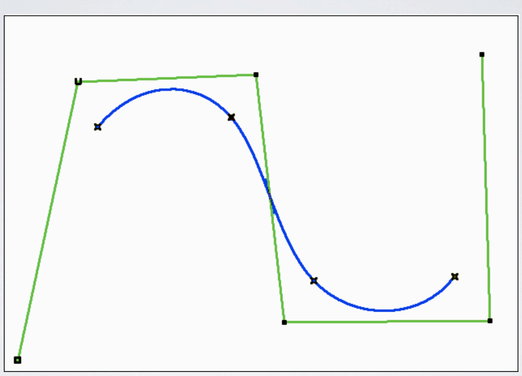




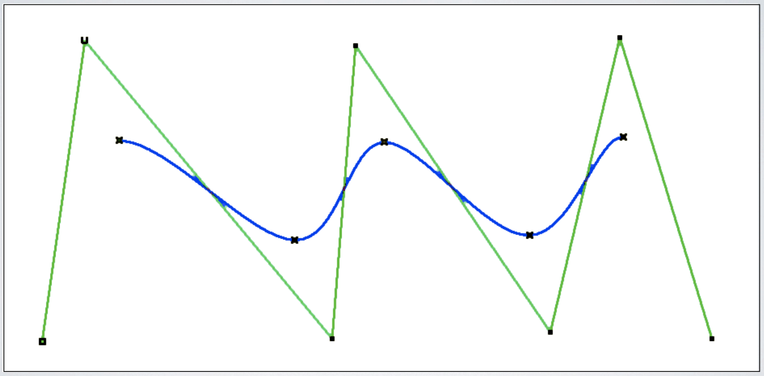
B-Splines



B-Splines

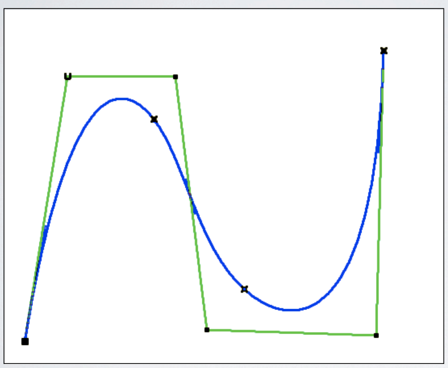


B-Splines



13

B-Splines



Example with end knots repeated

14

B-Splines

Build a curve w/ overlapping bumps

Continuity

- Inside bumps C^2
- Bumps "fade out" with C^2 continuity

Boundaries

- Circular (closed curves)
- Repeat end points
- Extra end points

15

B-Splines

Notation

- The basis functions are the $b_i(u)$
- "Hump" functions are the concatenated function
  - Sometimes the humps are called basis... can be confusing
- The u_i are the knot locations
- The weights on the hump/basis functions are control points

B-Splines

Similar construction for give higher continuity with higher degree polynomials

Repeating knots drops continuity

- Limit as knots approach each other

Still cubics, so conversion to other cubic basis is a matrix multiplication

B-Splines

Geometric construction

- Due to Cox and de Boor
- My own notation, beware if you compare w/ text

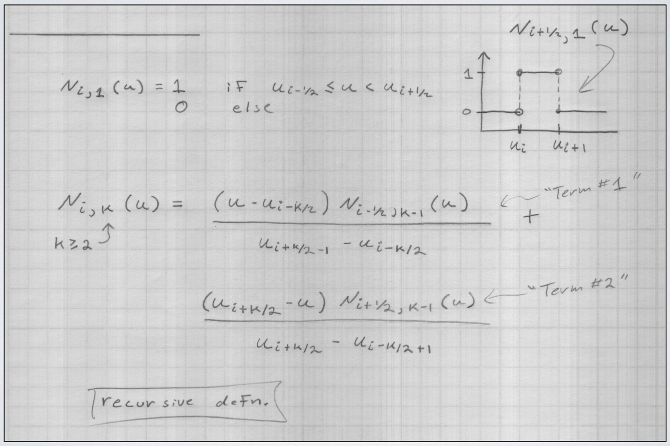
Let hump centered on u_i be $N_{i,4}(u)$

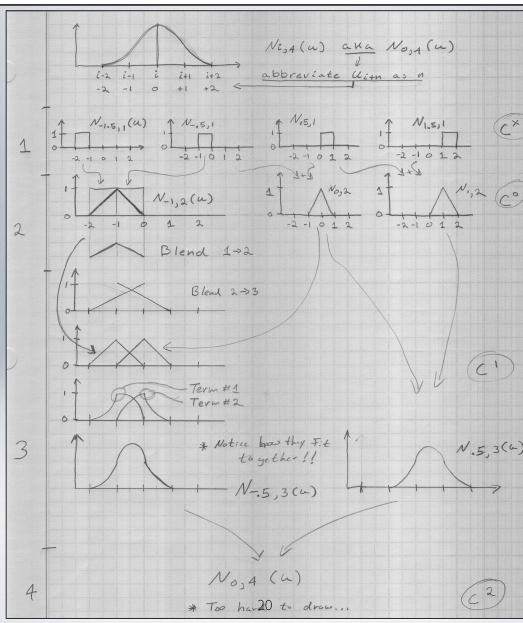
Cubic is order 4

$N_{i,k}(u)$ Is order k hump, centered at u_i

Note: i is integer if k is even
else $(i+1/2)$ is integer

B-Splines





NURBS

Non**u**niform **R**ational **B**-**S**plines

- Basically B-Splines using homogeneous coordinates
- Transform under perspective projection
- A bit of extra control

NURBS

$$\mathbf{p}_i = \begin{bmatrix} p_{ix} \\ p_{iy} \\ p_{iz} \\ p_{iw} \end{bmatrix}$$

$$\mathbf{x}(u) = \frac{\sum_i \begin{bmatrix} p_{ix} \\ p_{iy} \\ p_{iz} \end{bmatrix} N_i(u)}{\sum_i p_{iw} N_i(u)}$$

Non-linear in the control points

The p_{iw} are sometimes called "weights"