# CS-184: Computer Graphics

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

Prof. James O'Brien
University of California, Berkeley

1

## Natural Splines

- Draw a "smooth" line through several points

A real draftsman's spline.

Image from Carl de Boor's webpage.

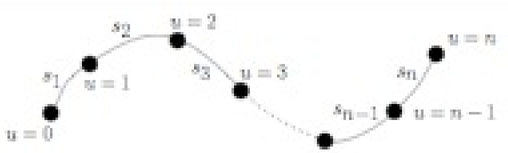
2

# 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...

3

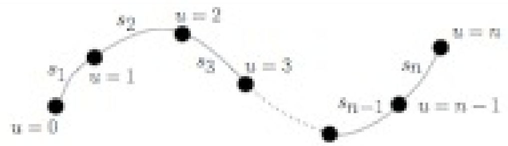
# Natural Cubic Splines




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

4

# Natural Cubic Splines



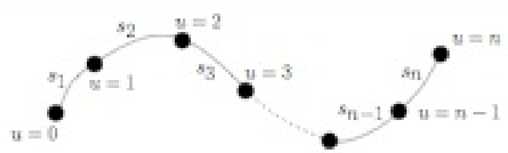

$$s_i(0) = p_{i-1} \qquad i = 1 \ldots n$$
$$s_i(1) = p_i \qquad i = 1 \ldots n$$

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

$$s_1''(0) = s_n''(1) = 0$$

5

# Natural Cubic Splines




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

$s_i(1) = p_i \qquad i = 1 \ldots n$ ← $n$ constraints

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

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

$s_1''(0) = s_n''(1) = 0$ ←2 constraints

**Total 4$n$ constraints**

5

## Natural Cubic Splines

- Interpolate data points
- No convex hull property
- Non-local support
  - Consider matrix structure...
- $C^2$ using cubic polynomials

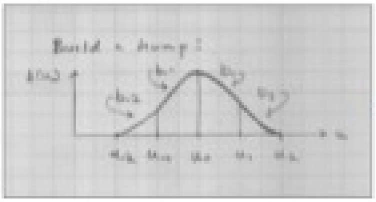
6

## B-Splines

- Goal: $C^2$ cubic curves with local support
  - Give up interpolation
  - Get convex hull property
- Build basis by designing "hump" functions

7

# B-Splines



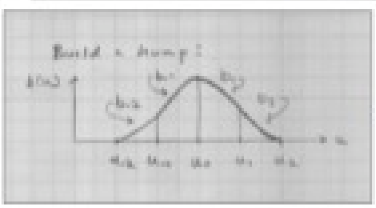

$$b(u) = \begin{cases} b_{-2}(u) & \text{if } u_{-2} \le u < u_{-1} \\ b_{-1}(u) & \text{if } u_{-1} \le u < u_0 \\ b_{+1}(u) & \text{if } u_0 \le u < u_{+1} \\ b_{+2}(u) & \text{if } u_{+1} \le u \le u_{+2} \end{cases}$$

$b''_{-2}(u_{-2}) = b'_{-2}(u_{-2}) = b_{-2}(u_{-2}) = 0$
$b''_{+2}(u_{+2}) = b'_{-2}(u_{+2}) = b_{-2}(u_{+2}) = 0$

$b_{-2}(u_{-1}) = b_{-1}(u_{-1})$
$b_{-1}(u_0) = b_{+1}(u_0)$
$b_{+1}(u_{+1}) = b_{+2}(u_{+1})$

8

# B-Splines




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

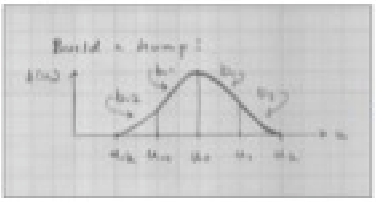
$b''_{-2}(u_{-2}) = b'_{-2}(u_{-2}) = b_{-2}(u_{-2}) = 0$ ← 3 constraints
$b''_{+2}(u_{+2}) = b'_{-2}(u_{+2}) = b_{-2}(u_{+2}) = 0$ ← 3 constraints

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

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

8

# B-Splines



$$b(u) = \begin{cases} b_{-2}(u) & \text{if } u_{-2} \le u < u_{-1} \\ b_{-1}(u) & \text{if } u_{-1} \le u < u_{0} \\ b_{+1}(u) & \text{if } u_{0} \le u < u_{+1} \\ b_{+2}(u) & \text{if } u_{+1} \le u \le u_{+2} \end{cases}$$

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

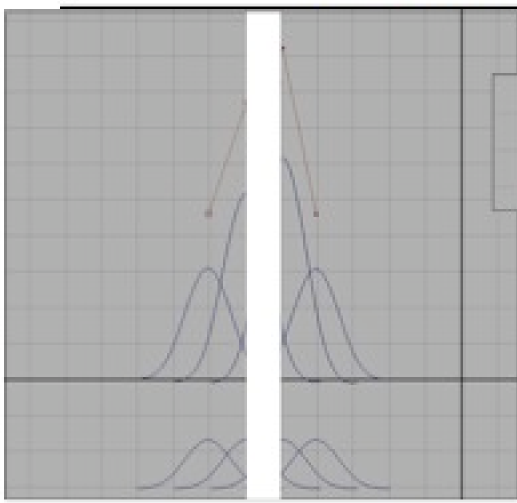
$b_{-2}(u_{-1}) = b_{-1}(u_{-1})$
$b_{-1}(u_0) = b_{+1}(u_0)$ ← [ Repeat for $b'$ and $b''$
$b_{+1}(u_{+1}) = b_{+2}(u_{+1})$   3×3=9 constraints

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

**Total 16 constraints**

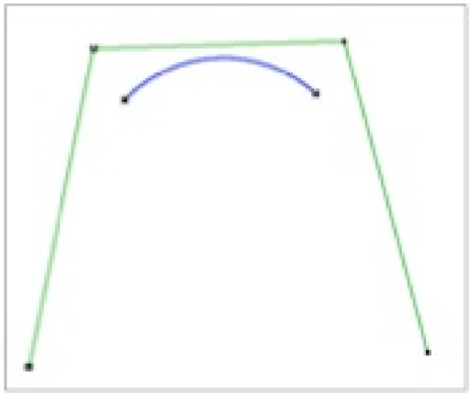
9

# B-Splines



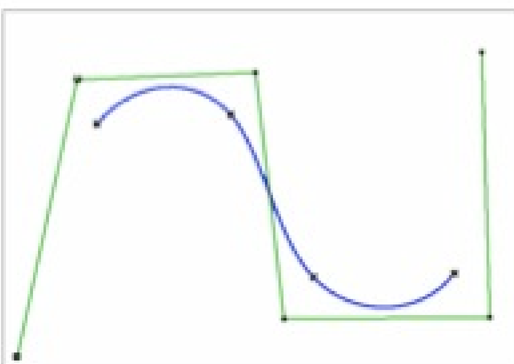
10

## B-Splines



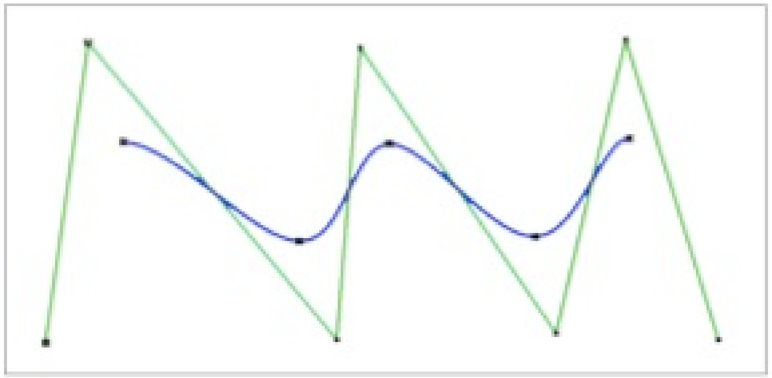
11

## B-Splines



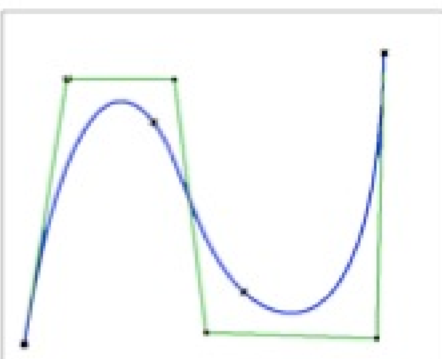
12

## 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
  - 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

16

## B-Splines

- Similar construction method can 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 just a matrix multiplication

17

## 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 $N_{i,4}(u)$

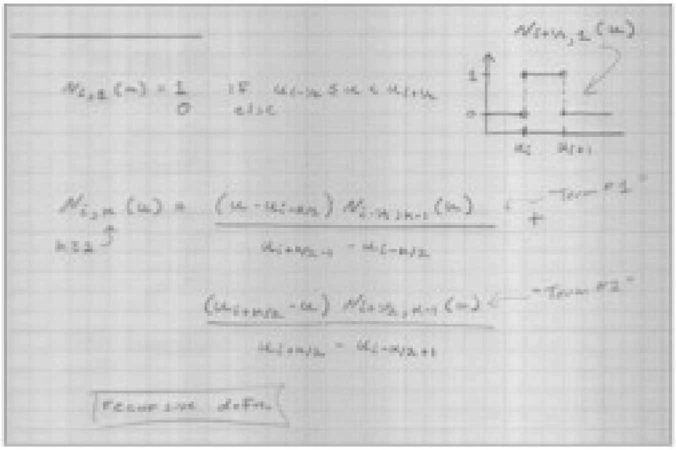
Cubic is order 4

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

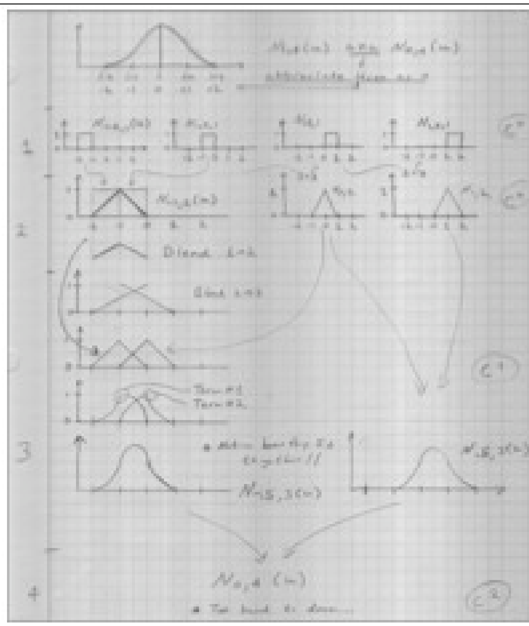
Note: $i$ is integer if $k$ is even
else $(i+1/2)$ is integer

18

## B-Splines





# NURBS

- **N**on**u**niform **R**ational **B**-**S**plines
  - Basically B-Splines using homogeneous coordinates
  - Transform under perspective projection
  - A bit of extra control

21

# 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"

22