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

\[ x(u) = \begin{cases} 
    s_1(u) & \text{if } 0 \leq u < 1 \\
    s_2(u - 1) & \text{if } 1 \leq u < 2 \\
    s_3(u - 2) & \text{if } 2 \leq u < 3 \\
    \vdots \\
    s_n(u - (n-1)) & \text{if } n-1 \leq u \leq n 
\end{cases} \]
Natural Cubic Splines

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

\[ s_i(1) = p_i, \quad i = 1 \ldots n \]  \(\leftarrow n \) constraints

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

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

\[ s_1''(0) = s_n''(1) = 0 \]  \(\leftarrow 2 \) 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

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

\[ b''(u_{-2}) = b'_{-2}(u_{-2}) = b_{-2}(u_{-2}) = 0 \quad \leftarrow 3 \text{ constraints} \]
\[ b''(u_{+2}) = b'_{+2}(u_{+2}) = b_{+2}(u_{+2}) = 0 \quad \leftarrow 3 \text{ constraints} \]
\[ b_{-2}(u_{-1}) = b_{-1}(u_{-1}) \]
\[ b_{-1}(u_{0}) = b_{+1}(u_{0}) \]
\[ b_{+1}(u_{+1}) = b_{+2}(u_{+1}) \quad \leftarrow \text{Repeat for } b' \text{ and } b'' \\
3 \times 3 = 9 \text{ constraints} \]

Total 15 constraints \ldots need one more
B-Splines

\[ b(u) = \begin{cases} 
   b_{-2}(u) & \text{if } u_2 \leq u < u_1 \\
   b_{-1}(u) & \text{if } u_1 \leq u < u_0 \\
   b_{+1}(u) & \text{if } u_0 \leq u < u_{+1} \\
   b_{+2}(u) & \text{if } u_{+1} \leq u \leq u_{+2} 
\end{cases} \]

\[
\begin{align*}
   b_{-2}'(u_2) &= b_{-2}'(u_2) = b_{-2}(u_2) = 0 \quad \leftarrow 3 \text{ constraints} \\
   b_{+2}'(u_{+2}) &= b_{+2}'(u_{+2}) = b_{+2}(u_{+2}) = 0 \quad \leftarrow 3 \text{ constraints} \\
   b_{-2}(u_1) &= b_{-1}(u_1) \\
   b_{-1}(u_0) &= b_{+1}(u_0) \\
   b_{+1}(u_{+1}) &= b_{+2}(u_{+1}) \quad \leftarrow \text{Repeat for } b' \text{ and } b'' \\
   3 \times 3 &= 9 \text{ constraints} \\
\end{align*}
\]

\[
\begin{align*}
   b_{-2}(u_2) + b_{-1}(u_1) + b_{+1}(u_0) + b_{+2}(u_{+1}) &= 1 \quad \leftarrow 1 \text{ constraint (convex hull)} \\
\end{align*}
\]

\textbf{Total 16 constraints}
B-Splines
B-Splines
B-Splines
B-Splines
B-Splines

Example with end knots repeated
B-Splines

- Build a curve with overlapping bumps
- Continuity
  - Inside bumps $C^2$
  - Bumps “fade out” with $C^2$ continuity
- Boundaries
  - Circular
  - Repeat end points
  - Extra end points
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 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
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
\[ N_{i,k}^j(u) = \begin{cases} 1 & \text{if } u_{i-k} \leq u < u_{i+k} \\ 0 & \text{else} \end{cases} \]

\[ N_{i,k}(u) = \frac{ (u-u_{i-k}) N_{i-k,k-1}(u) }{ u_{i+k-1} - u_{i-k} } + \frac{ (u_{i+k/2} - u) N_{i+k/2,k-1}(u) }{ u_{i+k/2} - u_{i-k/2+1} } \]

*Recursive defn.*
NURBS

• **Nonuniform Rational B-Splines**
  • Basically B-Splines using homogeneous coordinates
  • Transform under perspective projection
  • A bit of extra control
NURBS

\[ p_i = \begin{bmatrix} p_{ix} \\ p_{iy} \\ p_{iz} \\ p_{iw} \end{bmatrix} \]

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

- Non-linear in the control points
- The \( p_{iw} \) are sometimes called “weights”