B(asis)-Splines

Ashish Myles
CISE, UF
Splines

- Piecewise polynomial
- More flexible than single polynomials
  - can have finite support
  - can be periodic
- Degree $d$ splines – typically $C^{d-1}$ continuity
Some polynomial representations

- Polynomials
  - Power / Taylor series
  - Newton polynomials
  - Lagrange polynomials
  - Hermite polynomials
  - **Bézier** (very special case of B-spline)

- Splines
  - Box spline (for curves, same as uniform B-spline)
  - B-spline
B-spline examples

- http://www.cs.technion.ac.il/~cs234325/Applets/applets/bspline/GermanApplet.html
- http://www.doc.ic.ac.uk/~dfg/AndysSplineTutorial/BSplines.html
Polynomials as linear combinations

- Power basis \([1, t, t^2, \ldots, t^d]\)
  
  \[p(t) = \sum_{i=0}^{d} c_i t^i\]

- Taylor basis \([1, (t-t_0), (t-t_0)^2, \ldots, (t-t_0)^d]\)
  
  \[p(t) = \sum_{i=0}^{d} c_i (t-t_0)^i\]

- Bézier basis \([1-(1-t)^d, d(1-t)^{d-1} t, \ldots, \begin{pmatrix} d \cr i \end{pmatrix} (1-t)^{d-i} t^i, \ldots, t^d]\), terms in the binomial expansion of \(((1-t)+t)^d = 1\)
  
  \[p(t) = \sum_{i=0}^{d} c_i \begin{pmatrix} d \cr i \end{pmatrix} (1-t)^{d-i} t^i\]
Splines as linear combinations

- A linear combination of **spline** basis functions

- Defined by
  - $k$ knots $t_i$
    - non-decreasing sequence specifying domain
    - determines basis functions (hence continuity and ranges)
  - $n$ coefficients $c_i$
    - coefficients with which the basis functions are multiplied
  - degree $d$ automatically determined: $k = n + d + 1$
B-spline basis

- Basis function:
  \[ N_i^0(t) = \begin{cases} 1 & \text{if } t \in [t_i, t_{i+1}] \\ 0 & \text{otherwise} \end{cases} \]

  \[ N_i^d(t) = \frac{t-t_i}{t_{i+d}-t_i} N_i^{d-1}(t) + \frac{t_{i+d+1}-t}{t_{i+d+1}-t_{i+1}} N_{i+1}^{d-1}(t) \]

- Non-negative & finite support
- Shifts add up to 1 in their overlap
- (Uniform B-splines \(<=\) uniformly spaced knots)
Spline in B-spline form

- Curve (degree $d$): $p(t) = \sum_i c_i N_i^d(t)$

- Example: $c_i = [1, 3, 2, -1]$ (y-coord only)

- **Greville abscissa**: x-coord for $c_i$ (max of corresponding basis)

- $k = n + d + 1$ knots
Geometric Properties

- Curve (degree $d$): $p(t) = \sum_i c_i N_i^d(t)$

Affine invariance: $c_i' = A(c_i) \iff p'(t) = A(p(t))$

Convex hull property: curve lies within $CH(c_i)$
Variation diminishing property

- No line intersects the spline more times than it intersects the control polygon.

- i.e. The curve will not wiggle more than the control polygon.
An alternative to interpolation

- Interpolating samples suffers from the Gibb's phenomenon

- Treating samples as coefficients has no such problems – curve can't wiggle more than coefficients.
Examples of non-uniform splines

- Knot sequence can be denser in areas needing more degrees of freedom.
Decreasing inherent continuity

- **Knot multiplicity**
  - Repeating a knot $m$ times decreases the inherent continuity of the basis functions across the knot to $C^{d-m}$. Degree $d = 3$.

Same control points in all cases. Knot sequence is uniform except for multiplicity at $t = 7$. 
Evaluation – deBoor's Algorithm

- Evaluate at $t = 4.5$ by repeated knot insertion without changing the underlying function.
Convergence under knot insertion

- Repeated uniform knot insertion converges to function as fast as $O(h^2)$, with $h =$ knot width.
Derivatives

- Compute using divided differences
  - deg 1 lower
  - continuity 1 lower
  - domain the same

\[ a_i = \frac{d}{t_{i+d}-t_i} \left( c_i - c_{i-1} \right) \]
Matlab spline toolbox

- Written by deBoor himself
- I used for my figures:
  - spmak, spapi – create/interpolate a B-spline
  - fnplt – plot the B-spline
  - fnrfn – do knot insertion
  - fnnder, fnint – differentiation and integration
- It's well documented and comes with tutorials and demos
Summary

- The B-spline form is
  - geometrically intuitive
  - numerically robust
  - easy to differentiate
  - easy to make discontinuous
  - very, very knotty

- Matlab spline toolbox
More terms to look up

- Tensor-product B-splines
  - for surfaces, volumes
- Bézier
  - tensor-product (special case of B-splines)
  - total-degree (triangular) – no good B-spline equivalent
- Blossoms
  - Excellent theoretical tool
  - Inefficient for implementation, though
References

• Curves and Surfaces for CAGD
  – Gerald Farin

• Bézier and B-Spline Techniques
  – Hartmut Prautzsch, Wolfgang Boehm, and Marco Paluszny

• Matlab spline toolbox documentation & demos