Computer Graphics

Minho Kim

November 29, 2009

Table of contents

Outline

Representation of a line

How can we represent a line segment connecting $p_0 = (x_0, y_0)$ and $p_1 = (x_1, y_1)$?

1. *Explicit* representation

$$y = f(x) = mx + b = \frac{y_1 - y_0}{x_1 - x_0}x - \frac{x_0(y_1 - y_0)}{x_1 - x_0} + y_0,$$

   where $x_0 \le x \le x_1$.

2. *Implicit* representation

$$\begin{aligned} g(x, y) &= ax + by + c \\ &= (y_1 - y_0)x - (x_1 - x_0)y - x_0(y_1 - y_0) + y_0(x_1 - x_0) \\ &= 0 \end{aligned}$$

   where $x_0 \le x \le x_1$ and $y_0 \le y \le y_1$.

3. *Parametric* representation

$$\boldsymbol{p}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = (1 - u)\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} + u\begin{bmatrix} y_0 \\ y_1 \end{bmatrix}$$

   where $0 \le u \le 1$.

Representation of a 2D curve

1. Explicit
   - $y = f(x)$
   - "What is the value of y for a given x?"
2. Implicit
   - $g(x, y) = 0$
   - "Which (x, y) pairs satisfy the given equation?"
   - memebership
3. Parametric
   - $\boldsymbol{p}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} \in \mathbb{R}^2$
   - "How the point $(x(u), y(u))$ changes as the parameter u changes?"
   - $x(u)$ and $y(u)$ are both functions of u.
   - Mapping from 1D a real line to the curve.

Representation of a circle

How can we represent a circle with radius $r > 0$ centered at $(0, 0)$ in each representation?

1. Explicit representation (for upper half)

$$y = \sqrt{r^2 - x^2}$$

   where $-r \leq x \leq r$.

2. Implicit representation

$$x^2 + y^2 - r^2 = 0.$$

3. Parametric representation

$$\boldsymbol{p}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} \cos u \\ \sin u \end{bmatrix}$$

   where $0 \leq u < 2\pi$.

Parametric curves

- Easily extended to any dimension.

  ex) 3D curve: $\boldsymbol{p}(u) = \begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix}$

- Velocity

  $$\frac{d}{du}\boldsymbol{p}(u) = \begin{bmatrix} \frac{d}{du}x(u) \\ \frac{d}{du}y(u) \\ \frac{d}{du}z(u) \end{bmatrix}$$

Representation of a surface

1. Explicit representation
   $z = f(x, y)$
   Example:
   $z = \sqrt{r^2 - x^2 - y^2}$ (upper hemisphere)
2. Implicit representation
   $g(x, y, z) = 0$
   Example: $x^2 + y^2 + z^2 - r^2 = 0$ (sphere)
3. Parametric representation
   $$\boldsymbol{p}(t) = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix} \in \mathbb{R}^3$$
   Example: $\boldsymbol{p}(u, v) = \begin{bmatrix} 1-u & u \end{bmatrix} \begin{bmatrix} \boldsymbol{p}_{00} & \boldsymbol{p}_{01} \\ \boldsymbol{p}_{10} & \boldsymbol{p}_{11} \end{bmatrix} \begin{bmatrix} 1-v \\ v \end{bmatrix}$
   where $0 \leq u, v \leq 1$ and $\boldsymbol{p}_{00}, \boldsymbol{p}_{01}, \boldsymbol{p}_{10}, \boldsymbol{p}_{11} \in \mathbb{R}^3$. (bilinear patch)

Parametric surface

- How to find the normal vector at $\boldsymbol{p}(u,v)$

1. Tangential vector along parameter u:
$$\frac{\partial}{\partial u}\boldsymbol{p}(u,v) = \begin{bmatrix} \frac{\partial}{\partial u}x(u,v) \\ \frac{\partial}{\partial u}y(u,v) \\ \frac{\partial}{\partial u}z(u,v) \end{bmatrix}$$
2. Tangential vector along parameter v:
$$\frac{\partial}{\partial v}\boldsymbol{p}(u,v) = \begin{bmatrix} \frac{\partial}{\partial v}x(u,v) \\ \frac{\partial}{\partial v}y(u,v) \\ \frac{\partial}{\partial v}z(u,v) \end{bmatrix}$$
3. The tangent plane is defined by two vectors $\partial\boldsymbol{p}/\partial u$ and $\partial\boldsymbol{p}/\partial v$, therefore the normal vector at $\boldsymbol{p}(u,v)$ is

$$\boldsymbol{n}(u,v) = \frac{\partial}{\partial u}\boldsymbol{p}(u,v) \times \frac{\partial}{\partial v}\boldsymbol{p}(u,v)$$

Parametric curves

$$\boldsymbol{p}(u) = \sum_{j=0}^{n} \boldsymbol{p}_j b_j(u), \quad 0 \leq u \leq 1$$

- (Usually) flexible and robust
- $\{\boldsymbol{p}_j\}_{j=0}^{n}$ are called *control points*.
- $\{b_j(u)\}_{j=0}^{n}$ are called *basis functions*.
- Whiat kind of function to choose for $b(u)$?

Outline

Design criteria

- Local control of shape
  Small changes of input should cause only small changes of output.
- Smoothness and continuity
  The curve should be smooth enough.
- Ability to evaluate derivatives
- Stability
  Evaluation of curves should be stable.
- Ease of rendering

Outline

Parametric polynomial curve

1. Now we chose parametric form of curves:

$$\boldsymbol{p}(u) = \sum_{j=0}^{n} \boldsymbol{p}_j b_j(u) = \boldsymbol{P}\boldsymbol{b}(u)$$

   where

$$\boldsymbol{P} := \begin{bmatrix} \boldsymbol{p}_0 & \cdots & \boldsymbol{p}_n \end{bmatrix}$$
$$\boldsymbol{b}(u) := \begin{bmatrix} b_0(u) & \cdots & b_n(u) \end{bmatrix}^T$$

2. What kind of basis function to choose?
   $\rightarrow$ polynomials
3. Which degree?
   $\rightarrow$ mostly two (quadratic) or three (cubic)
4. Which ‘basis’ of polynomial to use?
   4.1 Power basis
   4.2 Interpolating polynomial
   4.3 Hermite basis
   4.4 Bernstein-Bézier basis
   4.5 B-spline

#1. Polynomial curves with power basis

- $\boldsymbol{p}(u) = \sum_{j=0}^{n} \boldsymbol{p}_j u^j = \boldsymbol{P}\boldsymbol{u}$
  where

$$\boldsymbol{P} := \begin{bmatrix} \boldsymbol{p}_0 & \boldsymbol{p}_1 & \cdots & \boldsymbol{p}_{n-1} & \boldsymbol{p}_n \end{bmatrix}$$
$$\boldsymbol{u} := \begin{bmatrix} 1 & u & \cdots & u^{n-1} & u^n \end{bmatrix}^T.$$

- Defined by $n+1$ *control points* $\{\boldsymbol{p}_j\}_{j=0}^{n}$.
- Example (quadratic):

$$\boldsymbol{p}(u) = \begin{bmatrix} -1 \\ 1 \end{bmatrix} + \begin{bmatrix} 2 \\ -2 \end{bmatrix} u + \begin{bmatrix} 0 \\ 3 \end{bmatrix} u^2, \quad 0 \le u \le 1$$

- Non-intuitive relation between control points and the curve
- Non-local control
- Unstable evaluation
- Non-invariant under affine transformations

Outline

#2. Interpolating polynomial

- Define a polynomial curve such that

$$\boldsymbol{p}(k/n) = \boldsymbol{p}_k, \quad 0 \le k \le n.$$

  $\longrightarrow$ Intuitive relation between control points and the curve!
- How to find $\boldsymbol{b}(u)$?
  1. Let

$$\boldsymbol{p}(u) = \sum_{j=0}^{n} \boldsymbol{p}_j b(u) = \boldsymbol{P}\boldsymbol{b}(u) = \sum_{j=0}^{n} \boldsymbol{c}_j u^j = \boldsymbol{C}\boldsymbol{u}$$

  where

$$\boldsymbol{P} := \begin{bmatrix} \boldsymbol{p}_0 & \cdots & \boldsymbol{p}_n \end{bmatrix}.$$

#2. Interpolating polynomial (cont'd)

- 2. Set a system of equations

$$\begin{aligned}
\boldsymbol{P} &= \begin{bmatrix} \boldsymbol{p}(0) & \cdots & \boldsymbol{p}(1) \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{c}_0 & \cdots & \boldsymbol{c}_n \end{bmatrix} \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \\ 0 & \frac{1}{n} & \cdots & \frac{n-1}{n} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & (\frac{1}{n})^n & \cdots & (\frac{n-1}{n})^n & 1 \end{bmatrix} \\
&=: \boldsymbol{C}\boldsymbol{A}.
\end{aligned}$$

$\rightarrow \boldsymbol{p}(u) = \boldsymbol{P}\boldsymbol{b}(u) = \boldsymbol{C}\boldsymbol{u} = (\boldsymbol{P}\boldsymbol{A}^{-1})\boldsymbol{u} = \boldsymbol{P}(\boldsymbol{A}^{-1}\boldsymbol{u})$
$\rightarrow \boldsymbol{b}(u) = \boldsymbol{A}^{-1}\boldsymbol{u}.$

#2. Interpolating polynomial (cont'd)

- More intuitive relation than power basis form
- Local control, but not good
- Still not stable evaluation
- Invariant under affine transformations

Outline

#3. Hermite basis

- How about using different 'controls'?
  - Two ends points to be interpolated. ($\boldsymbol{p}_0$ and $\boldsymbol{p}_3$)
  - Two derivarives at each end points. ($\boldsymbol{m}_0$ and $\boldsymbol{m}_3$)

  $\rightarrow$ Cubic polynomial required.
- $\boldsymbol{p}(u) = \boldsymbol{P}\boldsymbol{b}(u) := \begin{bmatrix} \boldsymbol{p}_0 & \boldsymbol{p}_3 & \boldsymbol{m}_0 & \boldsymbol{m}_3 \end{bmatrix} \boldsymbol{b}(u)$.
- How to find $\boldsymbol{b}(u)$?
  1. Let $\boldsymbol{p}(u) = \boldsymbol{P}\boldsymbol{b}(u) = \boldsymbol{C}\boldsymbol{u}$.
  2. First,

  $$\boldsymbol{p}(0) = \boldsymbol{c}_0 = \boldsymbol{p}_0 \quad \text{and} \quad \boldsymbol{p}(1) = \boldsymbol{c}_0 + \boldsymbol{c}_1 + \boldsymbol{c}_2 + \boldsymbol{c}_3 = \boldsymbol{p}_3.$$

  3. Next, since

  $$\boldsymbol{p}'(u) = \boldsymbol{C} \begin{bmatrix} 0 & 1 & 2u & 3u^2 \end{bmatrix}^T,$$

  we get

  $$\boldsymbol{p}'(0) = \boldsymbol{c}_1 = \boldsymbol{m}_0 \quad \text{and} \quad \boldsymbol{p}'(1) = \boldsymbol{c}_1 + 2\boldsymbol{c}_2 + 3\boldsymbol{c}_3 = \boldsymbol{m}_3.$$

#3. Hermite basis (cont'd)

- (cont'd)
  4. Now we get a system of equation

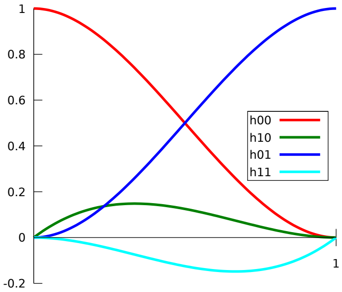
$$\begin{aligned} \boldsymbol{P} &= \begin{bmatrix} \boldsymbol{p}_0 & \boldsymbol{p}_3 & \boldsymbol{m}_0 & \boldsymbol{m}_3 \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{c}_0 & \boldsymbol{c}_1 & \boldsymbol{c}_2 & \boldsymbol{c}_3 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 3 \end{bmatrix} \\ &=: \boldsymbol{C}\boldsymbol{A}. \end{aligned}$$

$\rightarrow \boldsymbol{p}(u) = \boldsymbol{P}\boldsymbol{b}(u) = \boldsymbol{C}\boldsymbol{u} = (\boldsymbol{P}\boldsymbol{A}^{-1})\boldsymbol{u} = \boldsymbol{P}(\boldsymbol{A}^{-1}\boldsymbol{u})$.

$\rightarrow$

$$\begin{aligned} \boldsymbol{b}(u) &= \boldsymbol{A}^{-1}\boldsymbol{u} \\ &= \begin{bmatrix} 1 & 0 & -3 & 2 \\ 0 & 0 & 3 & -2 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ u \\ u^2 \\ u^3 \end{bmatrix}. \end{aligned}$$

#3. Hermite basis (cont'd)



- Intuitive control
- Local control
- Invariant under affine transformations ($b_0(u) + b_1(u) \equiv 1$)

Outline

Bernstein polynomial

Bernstein polynomial (of degree n)

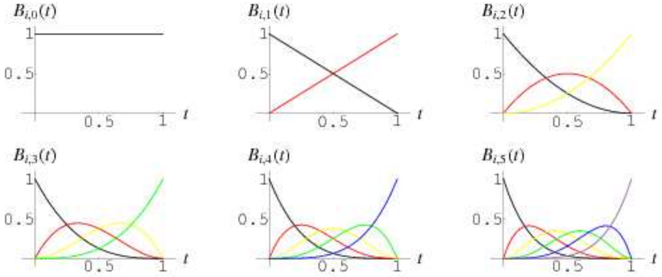
$$B_j^n(u) := \binom{n}{j} u^j (1-u)^{n-j}$$

where

$$\binom{n}{j} := \frac{n!}{j!(n-j)!}$$

are binomial coefficients.

Note that $B_j^n(u) \equiv 0$ for $j < 0$ or $j > n$.



Bernstein polynomial (cont'd)

Properties

- Recursive

$$B_j^n(u) = (1-u)B_j^{n-1}(u) + uB_{j-1}^{n-1}(u)$$

- All non-negative for $0 \le u \le 1$

$$B_j^n(u) \ge 0, \quad 0 \le j \le n, 0 \le u \le 1.$$

- Symmetry

$$B_j^n(u) = B_{n-j}^n(1-u).$$

- Partition of unity

$$\sum_{j=0}^{n} B_j^n(u) \equiv 1.$$

$\leftarrow$ binomial theorem

Bernstein polynomial (cont'd)

- Degree raising

$$(1-u)B_j^n(u) = \frac{n-j+1}{n+1}B_j^{n+1}(u).$$

- Derivative

$$\frac{d}{du}B_j^n(u) = n(B_{j-1}^{n-1}(u) - B_j^{n-1}(u)).$$

- Subdivision

$$B_j^n(cu) = \sum_{k=0}^{n} B_j^k(c)B_k^n(u).$$

Bézier curves

$$\boldsymbol{p}(u) = \sum_{j=0}^{n} \boldsymbol{p}_j B_j^n(u)$$

where $B_j^n(u)$ are Bernstein polynomials of degree n.
Properties:

- $\boldsymbol{p}(0) = \boldsymbol{p}_0$ and $\boldsymbol{p}(1) = \boldsymbol{p}_n$ $\rightarrow$ "endpoint interpolation"
- $\boldsymbol{p}(u) \neq 0$ for $0 < u < 1$ $\rightarrow$ Control points except two end points are not interpolated.
- $\sum_{j=0}^{n} B_j^n(u) = 1$
  - "convex hull property"
  - Invariant under affine transformations
  - Entire curve lies in the convex hull of the *control polygon*
- $B_j^n(u) = B_{n-j}^n(1-u)$ $\rightarrow$ symmetry
- $\operatorname{argmax}_u B_j^n(u) = j/n$ $\rightarrow$ local control

Bézier curves (cont'd)

- $\displaystyle\sum_{j=0}^{n} \frac{j}{n} B_j^n(u) = u \rightarrow$ "linear precision"
- Derivative

$$\frac{d}{du}\boldsymbol{p}(u) = n \sum_{j=0}^{n-1} (\boldsymbol{p}_{j+1} - \boldsymbol{p}_j) B_j^{n-1}(u).$$

  - A degree $n-1$ Bézier curve with its control points as *vectors*.
  - $\frac{d}{du}\boldsymbol{p}(0) = n(\boldsymbol{p}_1 - \boldsymbol{p}_0)$ and $\frac{d}{du}\boldsymbol{p}(1) = n(\boldsymbol{p}_n - \boldsymbol{p}_{n-1})$
- Subdivision

$$\sum_{j=0}^{n} \left(\sum_{k=0}^{j} \boldsymbol{p}_k B_k^j(c) \right) B_j^n(u) = \sum_{j=0}^{n} \boldsymbol{p}_j B_j^n(cu), \quad 0 \le c \le 1.$$

Example: Cubic Bézier curve subdivision

Cubic Bézier curve subdivision ($c = 1/2$, $n = 3$)

Since

$$
\begin{aligned}
&B_0^0(1/2) = 1, \\
&B_0^1(1/2) = B_1^1(1/2) = 1/2, \\
&B_0^2(1/2) = B_2^2(1/2) = 1/4, \quad B_1^2(1/2) = 1/2, \\
&B_0^3(1/2) = B_3^3(1/2) = 1/8, \quad B_1^3(1/2) = B_2^3(1/2) = 3/8,
\end{aligned}
$$

$$
\begin{aligned}
\sum_{j=0}^{3} \boldsymbol{p}_j B_j^3(u/2) = &\boldsymbol{p}_0 B_0^3(u) \\
&+ \frac{1}{2}(\boldsymbol{p}_0 + \boldsymbol{p}_1) B_1^3(u) \\
&+ \frac{1}{4}(\boldsymbol{p}_0 + 2\boldsymbol{p}_1 + \boldsymbol{p}_2) B_2^3(u) \\
&+ \frac{1}{8}(\boldsymbol{p}_0 + 3\boldsymbol{p}_1 + 3\boldsymbol{p}_2 + \boldsymbol{p}_3) B_3^3(u).
\end{aligned}
$$

Evaluating Bézier curves: de Casteljau's algorithm

A Bézier curve (of degree n) $\boldsymbol{p}(u) := \boldsymbol{p}_0^n(u)$ can be evaluated by the following recursive form:

$$\boldsymbol{p}_j^r(u) = (1-u)\boldsymbol{p}_j^{r-1}(u) + u\boldsymbol{p}_{j+1}^{r-1}(u), \quad \begin{cases} r = 1, \cdots, n \\ j = 0, \cdots, n-r. \end{cases}$$

with $\boldsymbol{p}_j^0(u) := \boldsymbol{p}_j$ (the control points).
$\rightarrow$ "repeated linear interpolation"
Example (cubic Bézier curve):

$$\begin{array}{cccc}
\boldsymbol{p}_0^0(u) := \boldsymbol{p}_0 & & & \\
& \boldsymbol{p}_0^1(u) & & \\
\boldsymbol{p}_1^0(u) := \boldsymbol{p}_1 & & \boldsymbol{p}_0^2(u) & \\
& \boldsymbol{p}_1^1(u) & & \boldsymbol{p}_0^3(u) := \boldsymbol{p}(u) \\
\boldsymbol{p}_2^0(u) := \boldsymbol{p}_2 & & \boldsymbol{p}_1^2(u) & \\
& \boldsymbol{p}_2^1(u) & & \\
\boldsymbol{p}_3^0(u) := \boldsymbol{p}_3 & & &
\end{array}$$

$\rightarrow$ "triangular scheme"

Tensored Bézier Patch

A tensored Bézier patch of degree (n_u, n_v) with $(n_u + 1) \times (n_v + 1)$ control points

$$\begin{matrix} \boldsymbol{p}_{0,0} & \cdots & \boldsymbol{p}_{0,j} & \cdots & \boldsymbol{p}_{0,n_v} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \boldsymbol{p}_{i,0} & \cdots & \boldsymbol{p}_{i,j} & \cdots & \boldsymbol{p}_{i,n_v} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \boldsymbol{p}_{n_u,0} & \cdots & \boldsymbol{p}_{n_u,j} & \cdots & \boldsymbol{p}_{n_u,n_v} \end{matrix}$$

is defined as

$$\boldsymbol{p}(u, v) = \sum_{i=0}^{n_u} \sum_{j=0}^{n_v} \boldsymbol{p}_{i,j} B_i^{n_u}(u) B_j^{n_v}(v).$$

Tensored Bézier Patch (cont'd)

A tensored Bézier patch can be considered as a collection of infinite number of Bézier curves, i.e., at $u = u_0$,

$$\begin{aligned}\boldsymbol{p}(u_0, v) &= \sum_{i=0}^{n_u} \sum_{j=0}^{n_v} \boldsymbol{p}_{i,j} B_i^{n_u}(u_0) B_j^{n_v}(v) \\ &= \sum_{j=0}^{n_v} \left(\sum_{i=0}^{n_u} \boldsymbol{p}_{i,j} B_i^{n_u}(u_0) \right) B_j^{n_v}(v)\end{aligned}$$

Outline

Chapter 11: Curves and Surfaces

Motivation

"What if we need more flexibility?"

$\rightarrow$ With Bézier curves, we should either

1. raise the degree or $\rightarrow$ too smooth curves
2. connect pieces smoothly. $\rightarrow$ restriction required

$\rightarrow$ B-spline curves

Uniform cubic B-splines

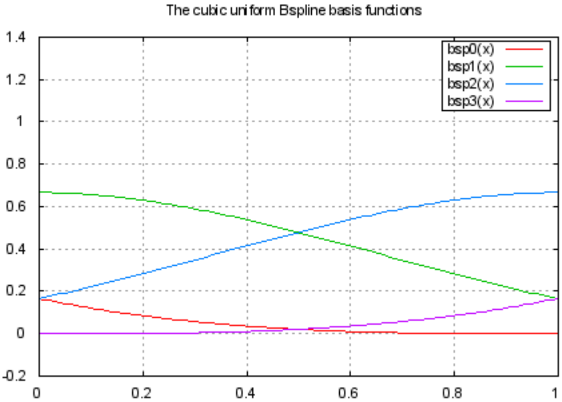
A B-spline with control points $\{\boldsymbol{p}_{-1}, \boldsymbol{p}_0, \boldsymbol{p}_1, \boldsymbol{p}_2\}$ is defined as

$$\boldsymbol{s}_0^3(u) := \sum_{j=0}^{3} \boldsymbol{p}_{-1+j} b_j^3(u), \quad u \in [0, 1)$$

where

$$\begin{aligned}
b_0^3(u) &:= \frac{1}{6}(1-u)^3 \\
b_1^3(u) &:= \frac{1}{6}(4 - 6u^2 + 3u^3) \\
b_2^3(u) &:= \frac{1}{6}(1 + 3u + 3u^2 - 3u^3) \\
b_3^3(u) &:= \frac{1}{6}u^3
\end{aligned}$$

Uniform cubic B-splines (cont'd)



(image courtesy of gnuplot)

Uniform cubic B-splines (cont'd)

More generally, the k-th segment with control points $\{\boldsymbol{p}_{k-1}, \boldsymbol{p}_k, \boldsymbol{p}_{k+1}, \boldsymbol{p}_{k+2}\}$ of a B-spline is defined as

$$\boldsymbol{s}_k^3(u) := \sum_{j=0}^{3} \boldsymbol{p}_{k-1+j} b_j^3(u-k), \quad u \in [k, k+1).$$

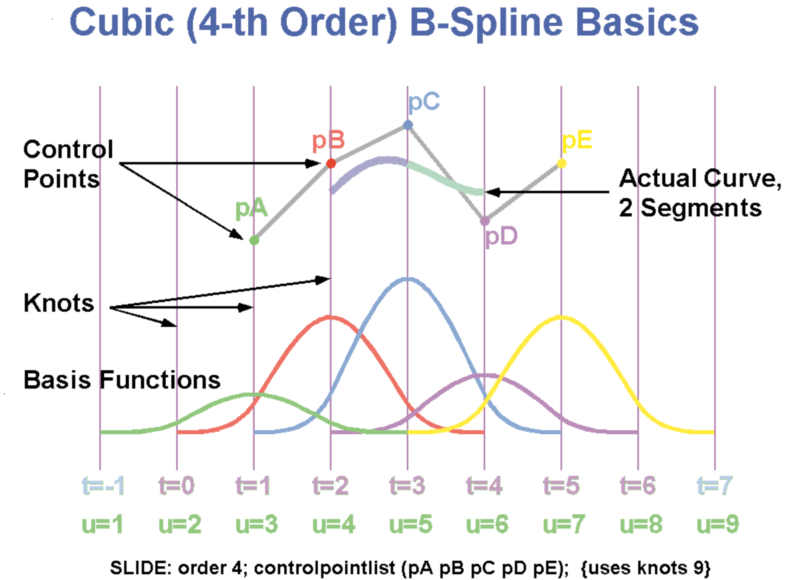
A B-spline on $u \in [0, m+1)$ is composed of the B-spline segments $\boldsymbol{s}_0^3(u), \cdots, \boldsymbol{s}_m^3(u)$ with control points $\{\boldsymbol{p}_{-1}, \cdots, \boldsymbol{p}_{m+2}\}$.

$$\boldsymbol{s}_{0,m}^3(u) := \sum_{j=-1}^{m+2} \boldsymbol{p}_j N^3(u-j)$$

where

$$N^3(u) := \begin{cases} b_3^3(u+2) & u \in [-2,-1) \\ b_2^3(u+1) & u \in [-1,0) \\ b_1^3(u) & u \in [0,1) \\ b_0^3(u-1) & u \in [1,2) \\ 0 & \text{otherwise.} \end{cases}$$

Uniform cubic B-splines (cont'd)



(image courtesy of Carlo H. Séquin at UC Berkeley)

Properties of uniform cubic B-spline basis functions

- Non-negativity: $N^3(u) \geq 0, \quad \forall u.$
- Local support

$$N^3(u) \begin{cases} > 0 & u \in [-2, 2) \\ = 0 & \text{otherwise.} \end{cases}$$

- Partition of unity

$$\sum_{j=-\infty}^{\infty} N^3(u) \equiv 1.$$

Properties of uniform cubic B-splines

- Invariant under affine transformations
- Convex hull property
- Control points not interpolated
- C^1 continuity

  At $u = k$,

$$\frac{d\boldsymbol{s}_{k-1}^3}{du}(k) = \sum_{j=0}^{3} \boldsymbol{p}_{k-2+j} \frac{db_j^3(u-k+1)}{du}\bigg|_{u=k}$$

$$= \sum_{j=0}^{3} \boldsymbol{p}_{k-2+j} \frac{db_j^3}{du}(1) = \frac{1}{2}(-\boldsymbol{p}_{k-1} + \boldsymbol{p}_{k+1})$$

and

$$\frac{d\boldsymbol{s}_{k}^3}{du}(k) = \sum_{j=0}^{3} \boldsymbol{p}_{k-1+j} \frac{db_j^3(u-k)}{du}\bigg|_{u=k}$$

$$= \sum_{j=0}^{3} \boldsymbol{p}_{k-1+j} \frac{db_j^3}{du}(0) = \frac{1}{2}(-\boldsymbol{p}_{k-1} + \boldsymbol{p}_{k+1}).$$

Properties of uniform cubic B-splines (cont'd)

- C^2 continuity
  At $u = k$,
  $$\frac{d\boldsymbol{s}_{k-1}^3}{du}(k) = \boldsymbol{p}_k - 2\boldsymbol{p}_{k+1} + \boldsymbol{p}_{k+2}$$
  and
  $$\frac{d\boldsymbol{s}_k^3}{du}(k) = \boldsymbol{p}_k - 2\boldsymbol{p}_{k+1} + \boldsymbol{p}_{k+2}.$$

Outline

Chapter 11: Curves and Surfaces

Non-uniform B-spline curves

Given m *knots* $\{u_j\}_{j=0}^{m-1}$ with $u_0 \leq u_1 \leq \cdots \leq u_{m-1}$, a B-spline of degree n is defined as

$$\boldsymbol{s}^n(u) = \sum_{j=0}^{m-n-2} \boldsymbol{p}_j N_j^n(u), \quad u \in [u_n, u_{m-n-1}]$$

where the basis functions are defined recursively (Cox-de Boor recursion):

$$N_j^0(u) := \begin{cases} 1, & u_j \leq u \leq u_{j+1}; \\ 0, & \text{otherwise} \end{cases}$$

$$N_j^k(u) := \frac{u - u_j}{u_{j+k} - u_j} N_j^{k-1}(u) + \frac{u_{j+k+1} - u}{u_{j+k+1} - u_{j+1}} N_{j+1}^{k-1}(u).$$

Properties

Properties of basis function

- Non-negativity and local support:

$$N_j^n(u) = \begin{cases} > 0, & u \in [u_j, u_{j+n+1}) \\ = 0, & \text{otherwise.} \end{cases}$$

- Partition of unity: $\sum_{j=-\infty}^{\infty} N_j^n(u) \equiv 1.$

Properties of Non-uniform B-splnes curves

- Invariant under affine transformations
- Convel hull property
- C^{n-1} continuity (at the “joints”)
- Each point (on the curve) is “controlled by” (or “affected by”) $n+1$ control points.
- Each control point “affects” $n+1$ curve segments.

de Boor's algorithm: Evaluating B-splines

"How to evaluate $\boldsymbol{s}^n(x)$?"

1. Find u_l and u_{l+1} such that $x \in [u_l, u_{l+1}]$.
2. Set $\boldsymbol{d}_j^{[0]} := \boldsymbol{p}_j$ for $j = l - n, \cdots, l$.
3. Compute

$$\boldsymbol{d}_j^{[k]} = (1 - \alpha_j^k)\boldsymbol{d}_{j-1}^{[k-1]} + \alpha_j^k \boldsymbol{d}_j^{[k-1]}, \quad \begin{cases} k = 1, \cdots, n, \\ j = l - n + k, \cdots, l \end{cases}$$

   with

$$\alpha_j^k := \frac{x - u_j}{u_{j+n+1-k} - u_j}.$$

4. $\boldsymbol{s}^n(x) = \boldsymbol{d}_l^{[n]}$.

NURBS: Non-Uniform Rational B-Splines

Curves represented by rational polynomials.

$$\sum_j \boldsymbol{p}_j \frac{N_j^n w_j}{\sum_j N_j^n w_j} = \frac{\sum_j N_j^n w_j \boldsymbol{p}_j}{\sum_j N_j^n w_j}$$

More control than NUBS. (e.g., perfect circle)

Outline

Rendering Curves

1. Evaluates the polynomial expression at (arbitrary) sample points and render piecewise linear curve.
   - Horner's method for power form
   - de Casterljau's algorithm for Bézier curves
   - de Boor's algorithm for B-splines
2. Subdivision of control polygon to approximate the curve. (Bézier curves and B-splines)

Outline

Outline

Outline