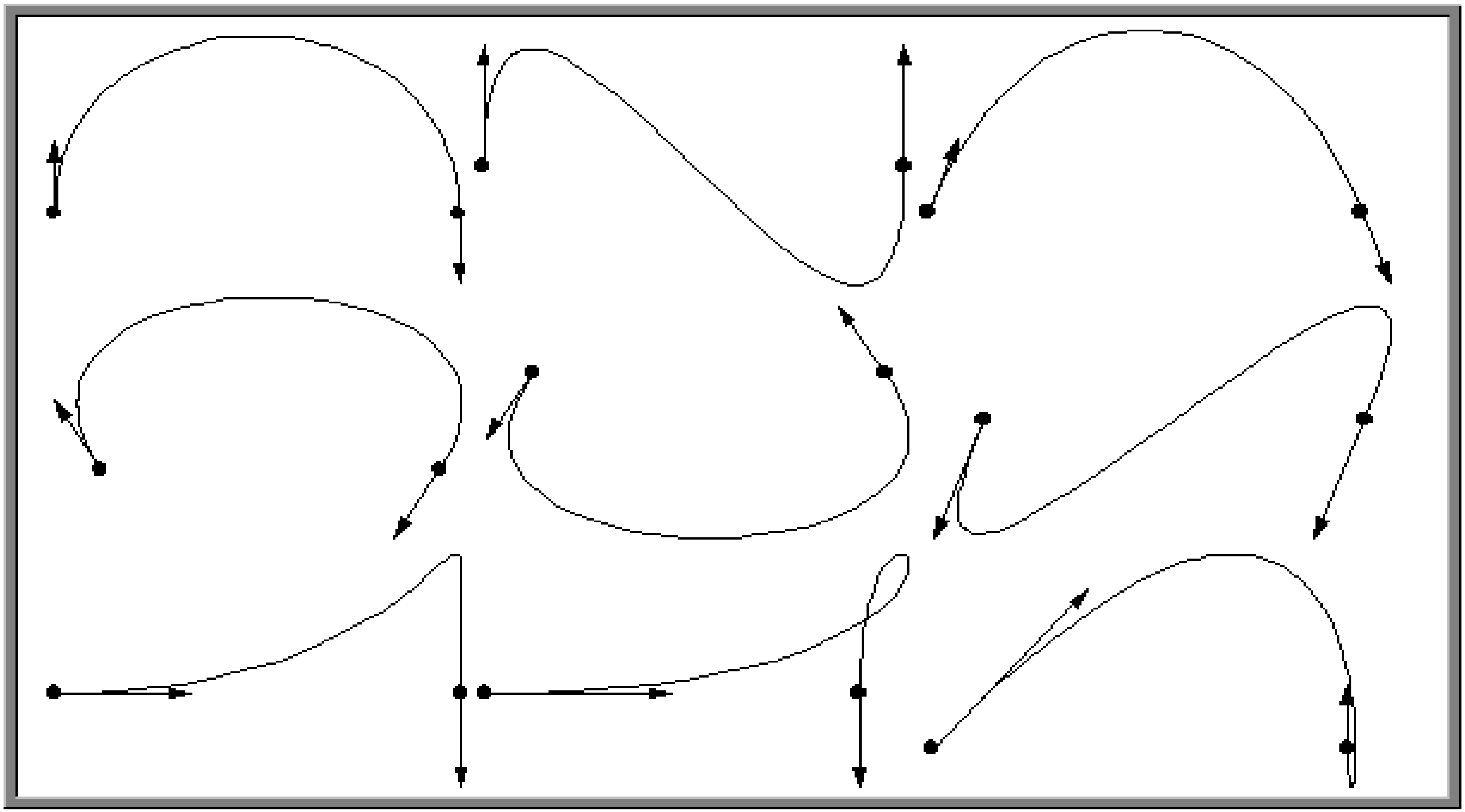
CS354 Computer Graphics
Surface Representation II

Qixing Huang
February 28th 2018

Review: Hermite Curve



Blending functions

- By multiplying first two matrices in lower-left equation, you have four functions of ‘t’ that blend the four control parameters

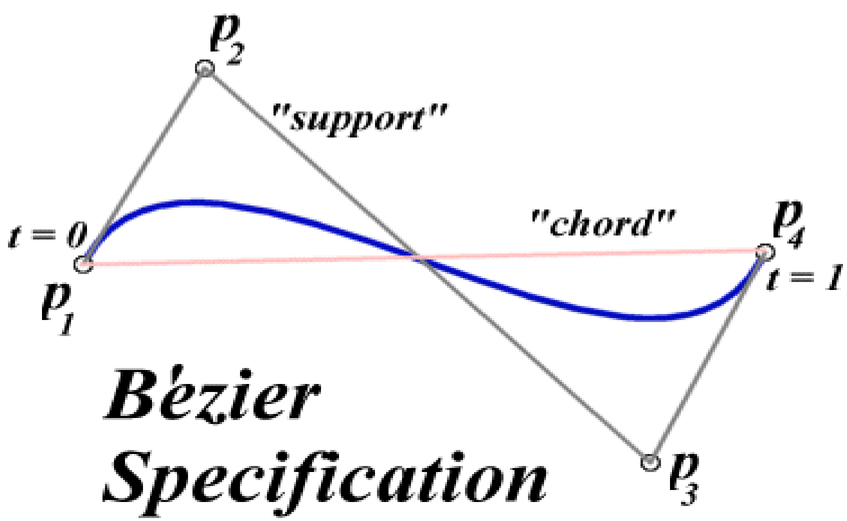
$$[x \quad y] = [t^3 \quad t^2 \quad t \quad 1] \underbrace{\begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{M}_{Hermite}} \underbrace{\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \frac{dx_1}{dt} & \frac{dy_1}{dt} \\ \frac{dx_2}{dt} & \frac{dy_2}{dt} \end{bmatrix}}_{\mathbf{G}_{Hermite}}$$

$$p(t) = \begin{bmatrix} 2t^3 - 3t^2 + 1 \\ -2t^3 + 3t^2 \\ t^3 - 2t^2 + t \\ t^3 - t^2 \end{bmatrix}^{\mathbf{T}} \begin{bmatrix} p_1 \\ p_2 \\ \nabla p_1 \\ \nabla p_2 \end{bmatrix}$$

Review: Bezier Curve

- Similar to Hermite, but more intuitive definition of endpoint derivatives

Four control points, two of which are knots



Review: Bézier vs. Hermite

- We can write our Bezier in terms of Hermite

$$\underbrace{\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \frac{dx_1}{dt} & \frac{dy_1}{dt} \\ \frac{dx_2}{dt} & \frac{dy_2}{dt} \end{bmatrix}}_{\mathbf{G}_{Hermite}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \end{bmatrix}}_{\mathbf{G}_{Bezier}}$$

Review: Bézier vs. Hermite

- The relation between polynomial coefficients and the constraints are linear

$$\begin{bmatrix} a_x & a_y \\ b_x & b_y \\ c_x & c_y \\ d_x & d_y \end{bmatrix} = \underbrace{\begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{M}_{Hermite}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \end{bmatrix}}_{\mathbf{G}_{Bezier}}$$

Bézier Curves

- Will always remain within bounding region defined by control points

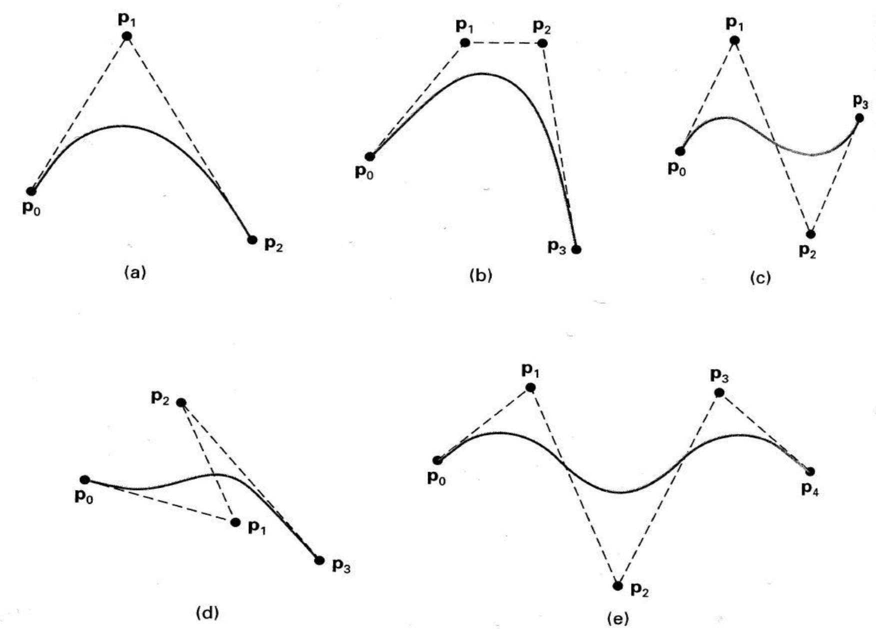


Figure 10-34
Examples of two-dimensional Bézier curves generated from three, four, and five control points. Dashed lines connect the control-point positions.

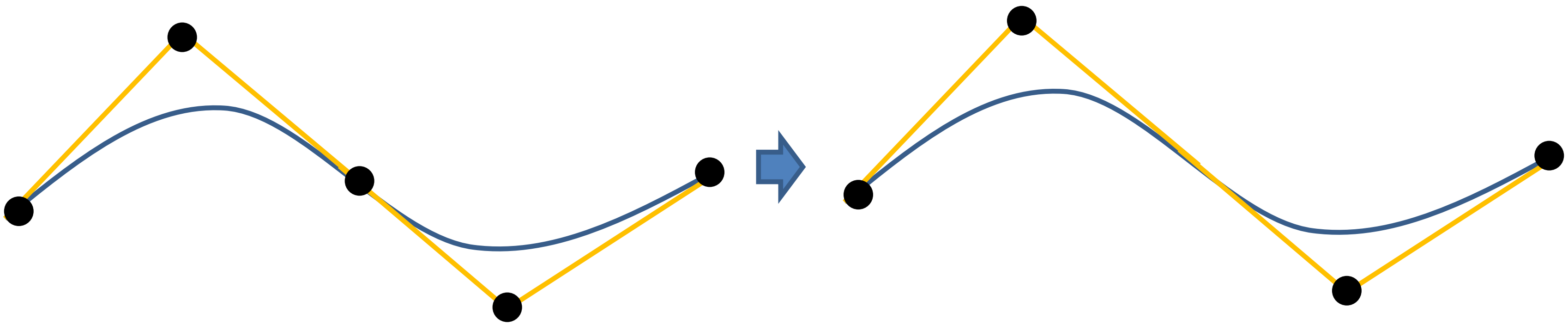
Limitations of Bezier Curves

- The degree of the polynomial is related to the number of control points

- No local control
  - Change one control point would change the entire curve

Bspline Curves

Motivating Example



Uniform Bspline

$$c_1(s) = \mathbf{p}_0(1-s)^2 + \mathbf{p}_1 2(1-s)s + \frac{\mathbf{p}_1 + \mathbf{p}_2}{2} s^2.$$

$$c_2(s) = \frac{\mathbf{p}_1 + \mathbf{p}_2}{2}(2-s)^2 + \mathbf{p}_2 2(2-s)(s-1) + \mathbf{p}_3(s-1)^2.$$

$$c(s) = \mathbf{p}_0 N_0(s) + \mathbf{p}_1 N_1(s) + \mathbf{p}_2 N_2(s) + \mathbf{p}_3 N_3(s).$$

Basis functions

$$N_0(s) = \begin{cases} (1-s)^2 & 0 \le s \le 1 \\ 0 & 1 \le s \le 2 \end{cases}$$

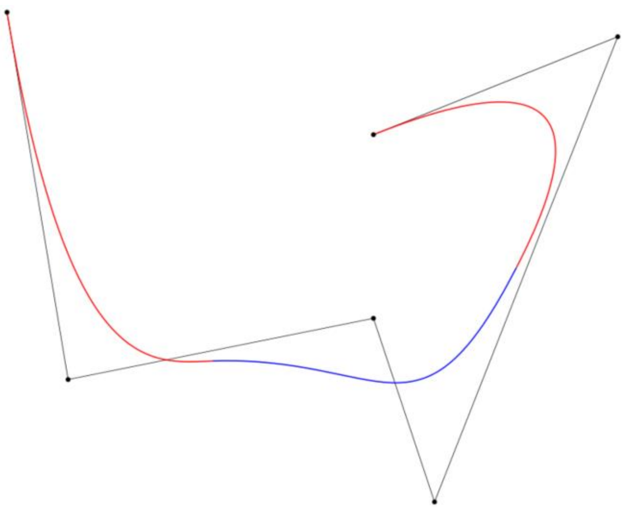
$$N_1(s) = \begin{cases} 2(1-s)s + \frac{s^2}{2} & 0 \le s \le 1 \\ \frac{(2-s)^2}{2} & 1 \le s \le 2 \end{cases}$$

$$N_2(s) = \begin{cases} \frac{s^2}{2} & 0 \le s \le 1 \\ \frac{(2-s)^2}{2} + 2(2-s)(s-1) & 1 \le s \le 2 \end{cases}$$

$$N_3(s) = \begin{cases} 0 & 0 \le s \le 1 \\ (s-1)^2 & 1 \le s \le 2 \end{cases}$$

Generalization to Bspline definition

- Control points
- Knot vector ***t** = (t₀, t₁, ... tₙ)*



$$N_{i,0}(t) := \begin{cases} 1 & \text{if } t_i \le t < t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,k}(t) := \frac{t - t_i}{t_{i+k} - t_i} N_{i,k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1}(t)$$

B-Spline Curve

- Bsplines are summarized from curves that stitch Bezier segments together
- Start with a sequence of control points
- Select four from middle of sequence (p_{i-2}, p_{i-1}, p_i, p_{i+1})
  - Bezier and Hermite goes between p_{i-2} and p_{i+1}
  - B-Spline doesn’t interpolate (touch) any of them but approximates the going through p_{i-1} and p_i

Uniform B-Splines

- Approximating Splines
- Approximates n+1 control points
  - $P_0, P_1, \ldots, P_n$, $n \geq 3$
- Curve consists of n –2 cubic polynomial segments
  - $Q_3, Q_4, \ldots Q_n$
- t varies along B-spline as Q_i: $t_i <= t < t_{i+1}$

- t_i (i = integer) are knot points that join segment Q_{i-1} to Q_i
- Curve is uniform because knots are spaced at equal intervals of parameter, t

Uniform B-Splines

- First curve segment, Q_3, is defined by first four control points

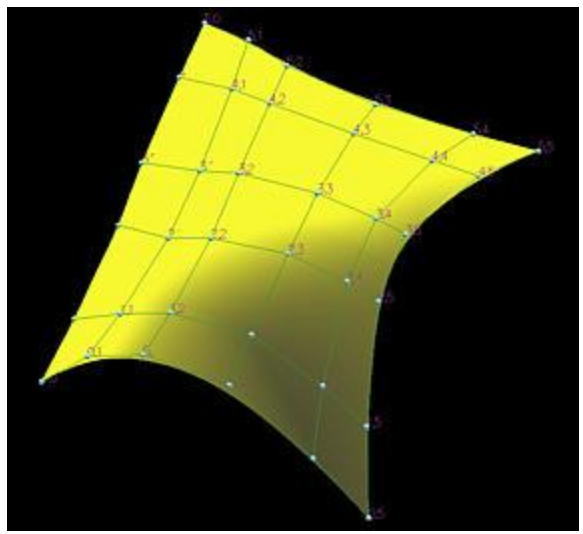
- Last curve segment, Q_m, is defined by last four control points, P_{m-3}, P_{m-2}, P_{m-1}, P_m

- Each control point affects four curve segments

Bspline Surfaces

Bspline Surfaces

- The same way to we generalize Bezier curves to Bezier surfaces



$$\mathbf{p}(u,v) = \sum_{i=0}^{m} \sum_{j=0}^{n} N_{i,p}(u) N_{j,q}(v) \mathbf{p}_{i,j}$$

NURBS Surfaces

NURBS Surfaces

- General form of a NURBS curve

$$C(u) = \sum_{i=1}^{k} \frac{N_{i,n} w_i}{\sum_{j=1}^{k} N_{j,n} w_j} \mathbf{P}_i = \frac{\sum_{i=1}^{k} N_{i,n} w_i \mathbf{P}_i}{\sum_{i=1}^{k} N_{i,n} w_i}$$

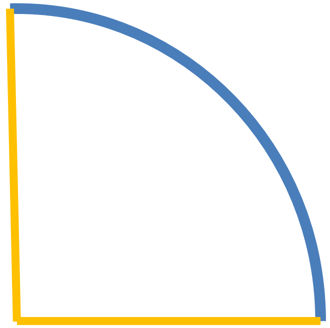
- Non-rational splines or Bezier curves may approximate a circle, but they cannot represent it exactly. Rational splines can represent any conic section, including the circle, exactly.

NURBS Representing an ARC

$$(x(u), y(u)) = \frac{w_0(1-u)^2(1,0) + w_1 2u(1-u)(1,1) + w_2 u^2(0,1)}{w_0(1-u)^2 + w_1 2u(1-u) + w_2 u^2}$$

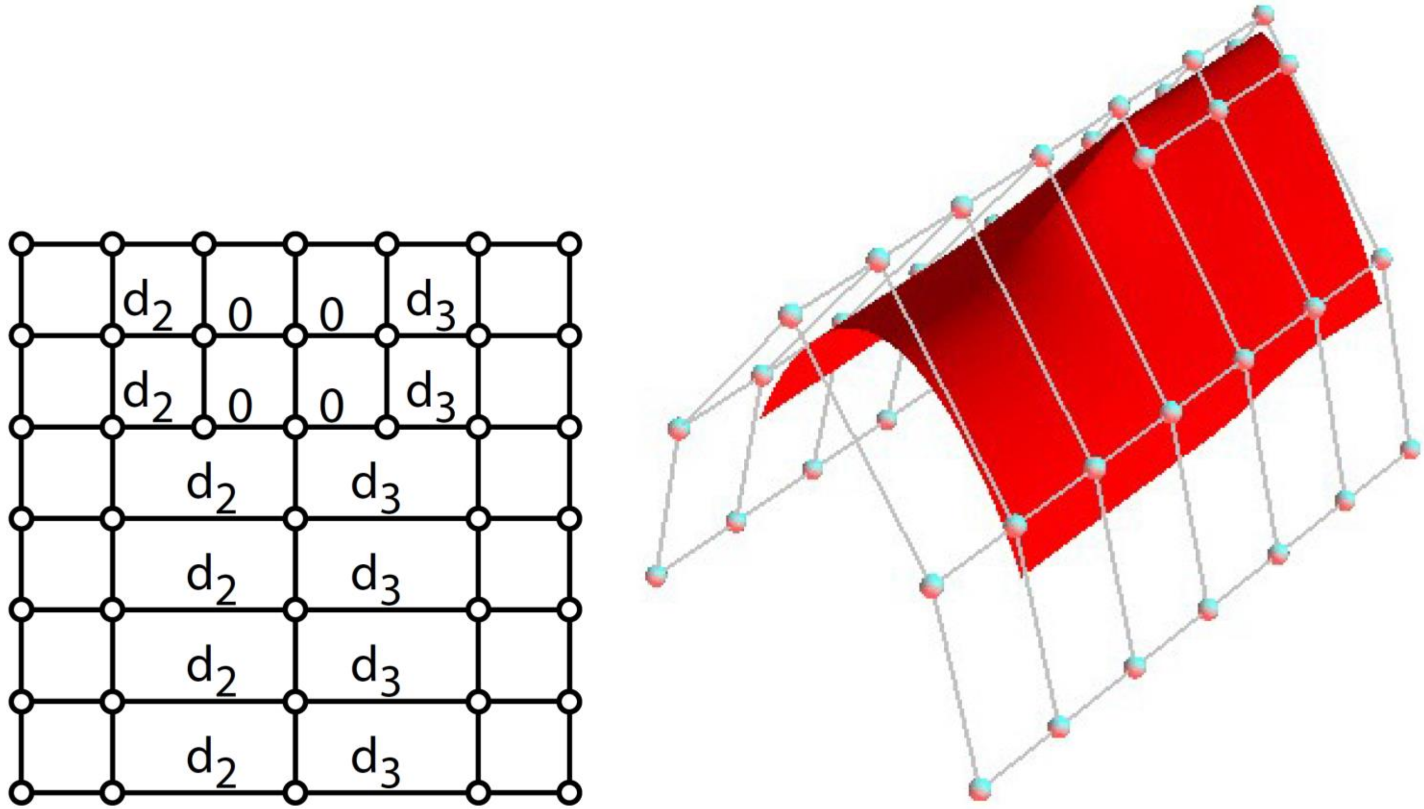
$$w_0 = 1,\ w_1 = 1,\ \text{and}\ w_2 = 2$$

$$(x(u), y(u)) = \frac{(1-u^2, 2u)}{1+u^2}$$



Tspline

TSpline



[Sederberg et al 03]

Questions?