

## Rational curves

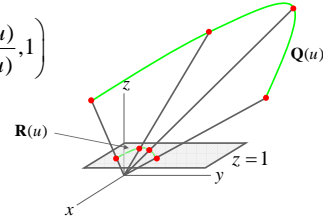
- A rational curve is a curve projected from a higher dimensional space. (4D→3D or 3D→2D)
- Adds shape flexibility to curve

72

## Projection of curves

- A 3D curve  $\mathbf{Q}(u)=(x(u),y(u),z(u))$  can be projected to a 2D curve  $\mathbf{R}$  on the plane  $z=1$  by dividing by  $z(u)$ :

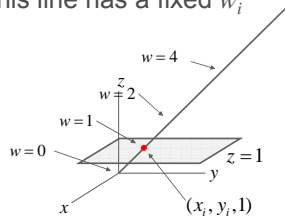
$$\mathbf{R}(u) = \left( \frac{x(u)}{z(u)}, \frac{y(u)}{z(u)}, 1 \right)$$



73

## Projection of curves

- A 2D point  $(x_i, y_i) \hat{=} (x_i, y_i, 1)$  can be extended into projection space as  $(w_i x_i, w_i y_i, w_i)$
- Each point on this line has a fixed  $w_i$



74

## Rational Bézier curve

- Suppose we have a Bézier curve  $\mathbf{Q}(u) = \sum_{i=0}^3 \mathbf{P}_i B_i(u)$
- With control points  $\mathbf{P}_i = (\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)$
- We can then consider the control points extensions of 2D points by writing them as

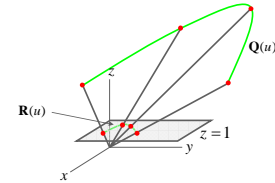
$$\mathbf{P}_i = (w_i x_i, w_i y_i, w_i)$$

- So

$$\mathbf{Q}(u) = \sum \begin{pmatrix} w_i x_i \\ w_i y_i \\ w_i \end{pmatrix} B_i(u)$$

- And we can project  $\mathbf{Q}$  to

$$\mathbf{R}(u) = \left( \frac{\sum w_i x_i B_i(u)}{\sum w_i B_i(u)}, \frac{\sum w_i y_i B_i(u)}{\sum w_i B_i(u)} \right) = \frac{\sum w_i \mathbf{P}_i^{2D} B_i(u)}{\sum w_i B_i(u)}$$



75

## Rational Bézier curve

- In this formulation

$$\mathbf{R}(u) = \frac{\sum w_i \mathbf{P}_i B_i(u)}{\sum w_i B_i(u)}$$

we see that  $w_i$  acts as a weight for  $\mathbf{P}_i$

- So the projection distance  $w_i (=z$  coordinate) of the control points acts as a control point weight for the 2D rational curve.

76

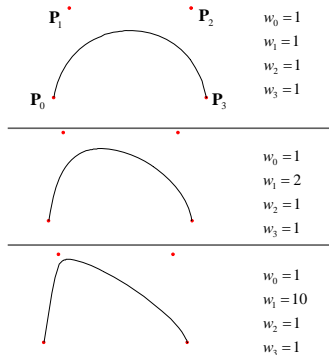
## Rational Bézier curve

$$\mathbf{R}(u) = \frac{\sum w_i \mathbf{P}_i B_i(u)}{\sum w_i B_i(u)}$$

- For 3D rational curves, the reasoning is similar (same formula at the end).
- The setup of rational curves as projections of a higher dimensional curve is primarily a useful mathematical/conceptual tool:
- For *drawing* the curve you need only a 'standard' set of control points and a set of weights

77

## Rational Bézier curve: example



$$\mathbf{R}(u) = \frac{\sum w_i \mathbf{P}_i B_i(u)}{\sum w_i B_i(u)}$$

Increasing the weight of a control point gives it more influence and draws the curve towards it

78

## Rational Bézier curve

- Moving a control points versus weight increase

79

## Rational Bézier curve

- Advantage of rational Bézier curve over the standard non-rational Bézier curve:  
Increased flexibility up to the point that conic sections (circles!) can now also be represented.

80

## NURBS

- NURBS = Non-uniform rational B-spline
- Maya
- Rational extension of B-spline (analogous to Bézier):

$$\mathbf{R}(u) = \frac{\sum w_i \mathbf{P}_i B_{i,d}(u)}{\sum w_i B_{i,d}(u)}$$

- Can be considered as a 'standard' non-uniform B-spline with added set of weights  $w_i$  for the control points

81

## NURBS

- Most popular representation in CAD
- Three parameter sets:
  - Control points
  - Knots
  - Weights
- Complete control of curve requires complete control of all three sets
- Alternative in some design programs: complete control of control points and a set of geometric primitives

82

## Surfaces

- Cubic curve segments extend to cubic surface patches in a straightforward manner:
- E.g. a cubic Bézier patch:

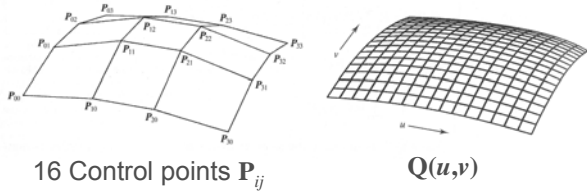
$$\mathbf{Q}(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 \mathbf{P}_{ij} B_i(u) B_j(v)$$

$0 \leq u, v \leq 1$

83

## Cubic Bézier patch

$$\mathbf{Q}(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 \mathbf{P}_{ij} B_i(u) B_j(v)$$



84

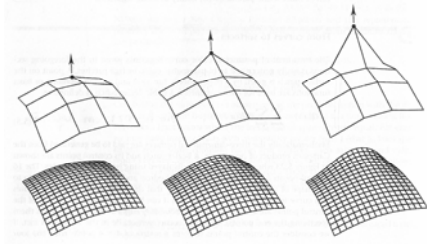
## Cubic Bézier patch

- Only the four corner points interpolate the surface
- The boundary curves are Bézier curves formed by the 4 pertinent control points
- The four interior points determine the interior shape

85

## Cubic Bézier patch

- Surface remains in convex hull of control polyhedron
- Intuitive editing (like with curves)



86

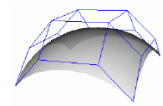
## Cubic Bézier patch

$$\mathbf{Q}(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 \mathbf{P}_{ij} B_i(u) B_j(v)$$

Full matrix form:

$$\mathbf{Q}(u, v) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \mathbf{B}_z \mathbf{P} \mathbf{B}_z^T$$

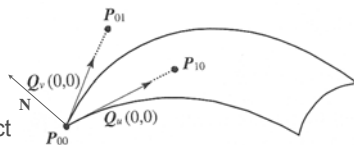
$$\mathbf{B}_z = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_{00} & \mathbf{P}_{01} & \mathbf{P}_{02} & \mathbf{P}_{03} \\ \mathbf{P}_{10} & \mathbf{P}_{11} & \mathbf{P}_{12} & \mathbf{P}_{13} \\ \mathbf{P}_{20} & \mathbf{P}_{21} & \mathbf{P}_{22} & \mathbf{P}_{23} \\ \mathbf{P}_{30} & \mathbf{P}_{31} & \mathbf{P}_{32} & \mathbf{P}_{33} \end{bmatrix}$$



87

## Cubic Bézier patch

- Tangent vectors: from partial derivatives
- Normal vectors: from vector product of tangents:



$$\mathbf{N} = \mathbf{Q}_u \times \mathbf{Q}_v \quad \mathbf{Q}_u(u, v) = \begin{bmatrix} 3u^2 & 2u & 1 & 0 \end{bmatrix} \mathbf{B}_z \mathbf{P} \mathbf{B}_z^T$$

$$\mathbf{Q}_v(u, v) = \begin{bmatrix} v^3 \\ v^2 \\ v \\ 1 \end{bmatrix}$$

$$\mathbf{Q}_u(u, v) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \mathbf{B}_z \mathbf{P} \mathbf{B}_z^T$$

Note that tangents at corners are along polyhedron edge (just like with Bézier curves)

88

## Cubic Bézier patch

- The cross-derivative  $\mathbf{Q}_{uv}$  is a measure for the rate of change of the tangent vector
- This a measure for surface *twist* (hence  $\mathbf{Q}_{uv}$  is also known as the twist vector)
- $\mathbf{Q}_{uv}$  is perpendicular to the surface

89

## Exercise

- Compute  $Q_{uv}$  for  $u=v=0$  for a cubic Bézier patch

$$Q(u, v) = [u^3 \ u^2 \ u \ 1] \mathbf{B}_z \mathbf{P} \mathbf{B}_z^T$$

$$\mathbf{B}_z = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_{00} & \mathbf{P}_{01} & \mathbf{P}_{02} & \mathbf{P}_{03} \\ \mathbf{P}_{10} & \mathbf{P}_{11} & \mathbf{P}_{12} & \mathbf{P}_{13} \\ \mathbf{P}_{20} & \mathbf{P}_{21} & \mathbf{P}_{22} & \mathbf{P}_{23} \\ \mathbf{P}_{30} & \mathbf{P}_{31} & \mathbf{P}_{32} & \mathbf{P}_{33} \end{bmatrix}$$

## Solution

- Fill in  $u=v=0$

$$Q_{uv}(0,0) = [0 \ 0 \ 1 \ 0] \mathbf{B}_z \mathbf{P} \mathbf{B}_z^T$$

$$= [-3 \ 3 \ 0 \ 0] \mathbf{P}$$

$$= [-3\mathbf{P}_{00} + 3\mathbf{P}_{10} \quad -3\mathbf{P}_{01} + 3\mathbf{P}_{11} \quad -3\mathbf{P}_{02} + 3\mathbf{P}_{12} \quad -3\mathbf{P}_{03} + 3\mathbf{P}_{13}]$$

$$= 9(\mathbf{P}_{00} - \mathbf{P}_{10} - \mathbf{P}_{01} + \mathbf{P}_{11})$$

## Solution

- First we compute the general derivatives

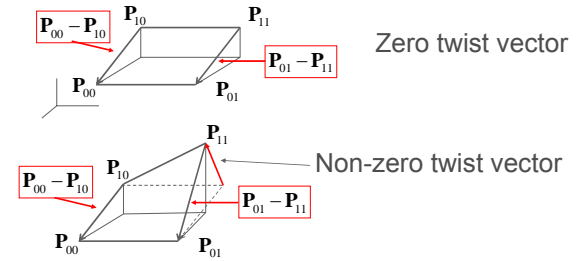
$$Q(u, v) = [u^3 \ u^2 \ u \ 1] \mathbf{B}_z \mathbf{P} \mathbf{B}_z^T$$

$$Q_u(u, v) = [3u^2 \ 2u \ 1 \ 0] \mathbf{B}_z \mathbf{P} \mathbf{B}_z^T$$

$$Q_{uv}(u, v) = [3u^2 \ 2u \ 1 \ 0] \mathbf{B}_z \mathbf{P} \mathbf{B}_z^T$$

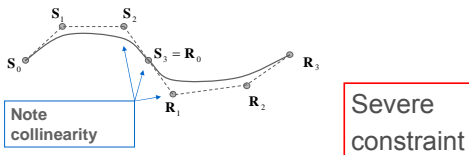
## Twist vector: geometric interpretation

$$9(\mathbf{P}_{00} - \mathbf{P}_{10} - \mathbf{P}_{01} + \mathbf{P}_{11}) = 9((\mathbf{P}_{00} - \mathbf{P}_{10}) - (\mathbf{P}_{01} - \mathbf{P}_{11}))$$

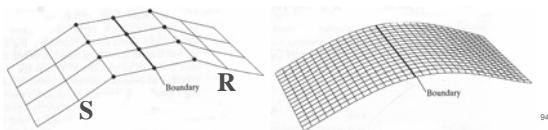


## Continuity of Bézier patches

The Bézier curve required 3 points to be collinear:

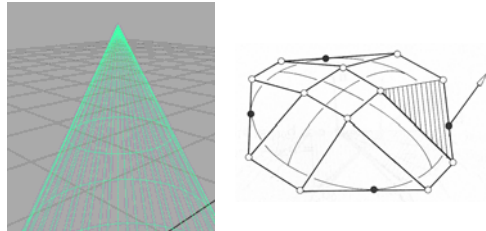


The patch requires 4 sets of 3 points to be collinear:



## Bézier patches

- Some things to watch out for:
  - The shapes that can be represented are limited (e.g. no spheres)
  - Patches may degenerate to triangles (possible normal vector ambiguity)



The Utah Teapot: Only 32 Patches!



96



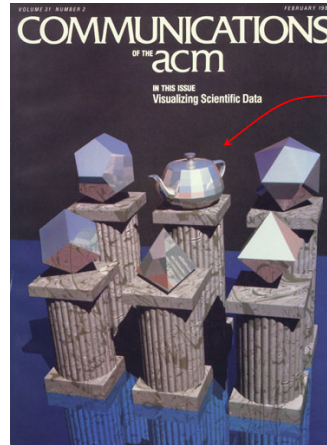
97



The 1974 original

The exhibit in the Boston Computer Museum

98

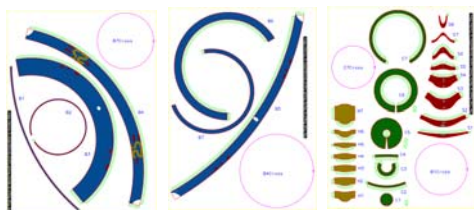


Real nerds know this as a 'teapotahedron'

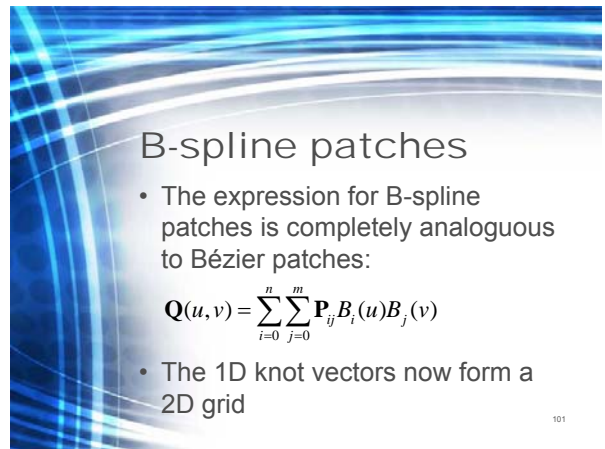
99

### The teapot obsession...

- Basic shape in 3ds max
- Pops up in the windows 'pipes' screensaver
- Pops up in 'Toy Story'
- Make your own teapot (SIGGRAPH 98)
- ...



100



### B-spline patches

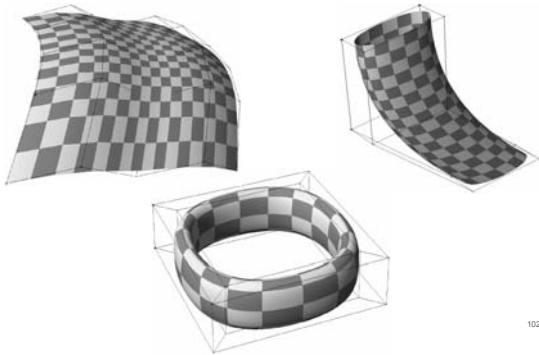
- The expression for B-spline patches is completely analogous to Bézier patches:

$$Q(u, v) = \sum_{i=0}^n \sum_{j=0}^m P_{ij} B_i(u) B_j(v)$$

- The 1D knot vectors now form a 2D grid

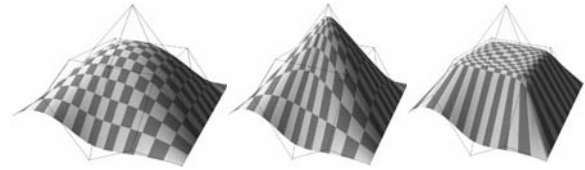
101

### Example: B-spline patches



102

### Example: NURBS patches



$$w = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$w = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 10 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$w = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 30 & 30 & 30 & 1 \\ 1 & 30 & 1 & 30 & 1 \\ 1 & 30 & 30 & 30 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

103



104