CS-C3100 Computer Graphics

Bézier Curves and Splines

3.2 Cubic Bézier Splines

Majority of slides from Frédo Durand
In This Video

• Cubic Bézier Curves: the prototype of a spline
• Manipulating polynomials with matrices
• A general formulation for polynomial splines
  – Not just Bézier: also Catmull-Rom, B-Splines, …
The Cubic Bézier Curve
Cubic Bézier Curve

- User specifies 4 control points $P_1 \ldots P_4$
- Curve goes through (interpolates) the ends $P_1, P_4$
- Approximates the two other ones
- Cubic polynomial
Cubic Bézier Curve

- 4 control points
- Curve passes through first & last control point
- Curve is tangent at $P_1$ to $(P_1-P_2)$ and at $P_4$ to $(P_4-P_3)$

A Bézier curve is bounded by the convex hull of its control points.
Cubic Bézier Curve

- 4 control points
- Curve passes through first & last control point
- Curve is tangent at $P_1$ to $(P_1 - P_2)$ and at $P_4$ to $(P_4 - P_3)$

A Bézier curve is bounded by the convex hull of its control points.
Cubic Bézier Curve

That is,

\[ x(t) = (1 - t)^3 x_1 + 3t(1 - t)^2 x_2 + 3t^2(1 - t) x_3 + t^3 x_4 \]

\[ y(t) = (1 - t)^3 y_1 + 3t(1 - t)^2 y_2 + 3t^2(1 - t) y_3 + t^3 y_4 \]
Cubic Bézier Curve

That is,

\[
\begin{align*}
\mathbf{P}(t) &= (1-t)^3 \mathbf{P}_1 + 3t(1-t)^2 \mathbf{P}_2 + 3t^2(1-t) \mathbf{P}_3 + t^3 \mathbf{P}_4 \\
y(t) &= (1-t)^3 y_1 + 3t(1-t)^2 y_2 + 3t^2(1-t) y_3 + t^3 y_4
\end{align*}
\]

Vectors with coordinates of control points

Scalar polynomials
Cubic Bézier Curve

- \( P(t) = (1-t)^3 P_1 + 3t(1-t)^2 P_2 + 3t^2(1-t) P_3 + t^3 P_4 \)

Verify what happens for \( t=0 \) and \( t=1 \)
“Bernstein Polynomials”

For cubic:

- $B_1(t) = (1-t)^3$
- $B_2(t) = 3t(1-t)^2$
- $B_3(t) = 3t^2(1-t)$
- $B_4(t) = t^3$

- (careful with indices, many authors start at 0)

- But defined for any degree
Properties of Bernstein polynomials

- $\geq 0$ for all $0 \leq t \leq 1$
- Sum to 1 for every $t$
  - called *partition of unity*
- (These two together are the reason why Bézier curves lie within convex hull)
- Only $B_1$ is non-zero at 0
  - Bezier interpolates $P_1$
  - Same for $B_4$ and $P_4$ for $t=1$
Interpretation as “Influence”

- Each $B_i$ specifies the influence of $P_i$
- First, $P_1$ is the most influential point, then $P_2$, $P_3$, and $P_4$
- $P_2$ and $P_3$ never have full influence
  - Not interpolated!
Bézier Curves, Concise Notation

- \( \mathbf{P}(t) = \mathbf{P}_1 B_1(t) + \mathbf{P}_2 B_2(t) + \mathbf{P}_3 B_3(t) + \mathbf{P}_4 B_4(t) \)
  - \( \mathbf{P}_i \) are 2D control points \((x_i, y_i)\)
  - For each \( t \), the point \( \mathbf{P}(t) \) on a Bézier curve is a linear combination of the control points with weights given by the Bernstein polynomials at \( t \)
Bézier Curves, Concise Notation

- \( P(t) = P_1B_1(t) + P_2B_2(t) + P_3B_3(t) + P_4B_4(t) \)
  - \( P_i \) are 2D control points \((x_i, y_i)\)
  - For each \( t \), the point \( P(t) \) on a Bézier curve is a linear combination of the control points with weights given by the Bernstein polynomials at

- Very nice, but only works for Bernstein polynomials. There are other splines too!
Basis for Cubic Polynomials

What’s a basis?

• A set of “atomic” vectors
  – Called basis vectors
  – Linear combinations of basis vectors span the space

• Linearly independent
  – Means that no basis vector can be obtained from the others by linear combination
    • Example: \( \mathbf{i}, \mathbf{j}, \mathbf{i}+\mathbf{j} \) is not a basis (missing \( \mathbf{k} \) direction!)

\[ \mathbf{v} = x \mathbf{i} + y \mathbf{j} + z \mathbf{k} \]
Canonical Basis for Cubics

\{1, t, t^2, t^3\}

- *Any* cubic polynomial is a linear combination of these: 
  \[ a_0 \times 1 + a_1 \times t + a_2 \times t^2 + a_3 \times t^3 \]
  - The *a*’s are the weights
- They are linearly independent
  - Means you can’t write any of the four monomials as a linear combination of the others. (You can try.)
Different basis

• For example:
  – \{1, 1+t, 1+2t+t^2, 1+t-t^2+6t^3\}
  – \{t^3, t^3+t^2, t^3+t, t^3+1\}

• These can all be obtained from \(1, t, t^2, t^3\) by linear combination
  – Just like all bases for Euclidean space can be obtained by linear combinations of the canonical \(i, j, \ldots\)

• Infinite number of possibilities, just like you have an infinite number of bases to span \(\mathbb{R}^2\)
Why we bother:
Matrix-Vector Notation For Polynomials

- For example:
  - $1$, $1+t$, $1+t+t^2$, $1+t-t^2+t^3$
  - $t^3$, $t^3+t^2$, $t^3+t$, $t^3+1$

These relationships hold for each value of $t$

\[
\begin{pmatrix} 1 \\ 1 + t \\ 1 + t + t^2 \\ 1 + t - t^2 + t^3 \end{pmatrix}
= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & -1 & 1 \end{pmatrix}
\begin{pmatrix} t \\ t^2 \\ t^3 \end{pmatrix}
\]

\[
\begin{pmatrix} t^3 \\ t^3 + t^2 \\ t^3 + t \\ t^3 + 1 \end{pmatrix}
= \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}
\begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix}
\]
Matrix-Vector Notation For Polynomials

• For example:
  - $1$, $1+t$, $1+t+t^2$, $1+t-t^2+t^3$
  - $t^3$, $t^3+t^2$, $t^3+t$, $t^3+1$

\[
\begin{pmatrix}
1 \\
1 + t \\
1 + t + t^2 \\
1 + t - t^2 + t^3
\end{pmatrix}
\begin{pmatrix}
1 \\
0 \\
0 \\
0
\end{pmatrix}
= 
\begin{pmatrix}
1 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 \\
1 & 1 & -1 & 1
\end{pmatrix}
\begin{pmatrix}
1 \\
t \\
t^2 \\
t^3
\end{pmatrix}
\]

“Canonical” monomial basis

Change-of-basis matrix

Not any matrix will do! If it’s singular, the basis set will be linearly dependent, i.e., redundant.
Matrix Form of Bernstein

Cubic Bernstein:

- \( B_1(t) = (1-t)^3 \)
- \( B_2(t) = 3t(1-t)^2 \)
- \( B_3(t) = 3t^2(1-t) \)
- \( B_4(t) = t^3 \)

Expand these out and collect powers of \( t \).

The coefficients are the entries in the matrix \( B \):

\[
\begin{bmatrix}
B_1(t) \\
B_2(t) \\
B_3(t) \\
B_4(t)
\end{bmatrix}
=
\begin{bmatrix}
1 & -3 & 3 & -1 \\
0 & 3 & -6 & 3 \\
0 & 0 & 3 & -3 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
t \\
t^2 \\
t^3
\end{bmatrix}
\]
Bézier in Matrix-Vector Notation

• Remember:
\[ P(t) = P_1 B_1(t) + P_2 B_2(t) + P_3 B_3(t) + P_4 B_4(t) \]
is a linear combination of control points

• or, in matrix-vector notation

\[
\begin{pmatrix}
   x(t) \\
   y(t)
\end{pmatrix}
= 
\begin{pmatrix}
   x_1 & x_2 & x_3 & x_4 \\
   y_1 & y_2 & y_3 & y_4
\end{pmatrix}
\begin{pmatrix}
   B_1(t) \\
   B_2(t) \\
   B_3(t) \\
   B_4(t)
\end{pmatrix}
\]

Bernstein polynomials (4x1 vector)

point on curve (2x1 vector)
matrix of control points (2 x 4)
Bézier in Matrix-Vector Notation

\[
\begin{pmatrix}
B_1(t) \\
B_2(t) \\
B_3(t) \\
B_4(t)
\end{pmatrix} =
\begin{bmatrix}
1 & -3 & 3 & -1 \\
0 & 3 & -6 & 3 \\
0 & 0 & 3 & -3 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{pmatrix}
1 \\
t \\
t^2 \\
t^3
\end{pmatrix}
\]

<= Flasback from two slides ago, let's combine with below:

Bernstein polynomials
(4x1 vector)

\[
P(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \end{pmatrix}
\begin{pmatrix}
B_1(t) \\
B_2(t) \\
B_3(t) \\
B_4(t)
\end{pmatrix}
\]

point on curve
(2x1 vector)

matrix of control points
(2 x 4)
Phase 3: Profit

- Combined, we get cubic Bézier in matrix notation

\[
P(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \end{pmatrix} \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix}
\]

- **Point on curve** (2x1 vector)
- **Geometry matrix** of control points \(P_1..P_4\) (2 x 4)
- **Spline matrix** (Bernstein)
- **Canonical monomial basis**
General Spline Formulation

\[ Q(t) = GBT(t) = \text{Geometry } G \cdot \text{Spline Basis } B \cdot \text{Power Basis } T(t) \]

- Geometry: control points coordinates assembled into a matrix \( G = (P_1, P_2, \ldots, P_{n+1}) \)
- Spline matrix \( B \): defines the type of spline
  - Bernstein for Bézier
- Power basis \( T \): the monomials \( (1, t, \ldots, t^n)^T \)
- Advantage of general formulation
  - Compact expression
  - Easy to convert between types of splines
  - Dimensionality (plane or space) doesn’t really matter
What can we do with this?

- Cubic polynomials form a vector space.
- Bernstein basis defines Bézier curves.
- Can do other bases as well: Catmull-Rom splines interpolate all the control points, for instance

\[
\mathbf{B}_{CR} = \frac{1}{2} \begin{pmatrix}
0 & -1 & 2 & -1 \\
2 & 0 & -5 & 3 \\
0 & 1 & 4 & -3 \\
0 & 0 & -1 & 1
\end{pmatrix}
\]
3D Spline has 3D control points

• The 2D control points can be replaced by 3D points – this yields space curves.
  – All the above math stays the same except with the addition of 3rd row to control point matrix (z coords)
  – In fact, can do homogeneous coordinates as well!
Linear Transformations & Cubics

- What if we want to transform each point on the curve with a linear transformation $\mathbf{M}$?

$$P'(t) = \mathbf{M} \begin{pmatrix} P_1,x & P_2,x & P_3,x & P_4,x \\ P_1,y & P_2,y & P_3,y & P_4,y \end{pmatrix} \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix}$$
Linear Transformations & Cubics

• What if we want to transform each point on the curve with a linear transformation $M$?
  – Because everything is linear, it’s the same as transforming the only the control points

$$P'(t) = M \begin{pmatrix} P_{1,x} & P_{2,x} & P_{3,x} & P_{4,x} \\ P_{1,y} & P_{2,y} & P_{3,y} & P_{4,y} \end{pmatrix} \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix} =$$

$$M \begin{pmatrix} P_{1,x} & P_{2,x} & P_{3,x} & P_{4,x} \\ P_{1,y} & P_{2,y} & P_{3,y} & P_{4,y} \end{pmatrix} \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix}$$
Linear Transformations & Cubics

• Homogeneous coordinates also work
  – Means you can translate, rotate, shear, etc.
  – Also, changing $w$ gives a “tension” parameter

• Note though that you need to normalize $P'$ by $1/w$

$$P'(t) = \mathbf{M} \begin{pmatrix} P_{1,x} & P_{2,x} & P_{3,x} & P_{4,x} \\ P_{1,y} & P_{2,y} & P_{3,y} & P_{4,y} \end{pmatrix} \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix}$$

$$= \mathbf{M} \begin{pmatrix} P_{1,x} & P_{2,x} & P_{3,x} & P_{4,x} \\ P_{1,y} & P_{2,y} & P_{3,y} & P_{4,y} \end{pmatrix} \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix}$$
Recap

- Splines turn control points into smooth curves
- Cubic polynomials form a vector space
- Bernstein basis gives rise to Bézier curves
  - Can be seen as influence function of data points
  - Or data points are coordinates of the curve in the Bernstein basis
- We can change between bases with matrices