Outline

1. Introduction to Curves
2. Parametric Curves
3. Properties of Bézier Curves
4. The Matrix Form
5. Derivatives
6. Composite Curves
7. The Geometry of Planar Curves
8. Moving along a Curve
9. WYSK
Focus of this chapter: cubic Bézier curves
Invented for car design
— France early 1960s at Rénault and Citroën in Paris
— Techniques still in use today
— called Geometric Modeling or Computer Aided Geometric Design

Apply linear algebra and geometric concepts to the study of curves
Parametric Curves

Straight line in *parametric* form:

\[
x(t) = (1 - t)a + tb \quad \Rightarrow \quad x(0) = a \text{ and } x(1) = b
\]

Interpret \( t \) as time and \( x(t) \) as a moving point

Coefficients \((1 - t)\) and \( t \) are linear polynomials \(\Rightarrow\) *linear interpolation*

A *parametric curve*: a curve that can be written as

\[
x(t) = \begin{bmatrix} f(t) \\ g(t) \end{bmatrix}
\]

where \( f(t) \) and \( g(t) \) are functions of the parameter \( t \)

Linear interpolant: \( f(t) = (1 - t)a_1 + tb_1 \) and \( g(t) = (1 - t)a_2 + tb_2 \)

In general: \( f \) and \( g \) can be any functions — polynomial, trig, exponential, ...
Next: study motion along curves

Cubic Bézier curves:
Start with four points in 2D or 3D

\[ b_0, b_1, b_2, b_3 \]

Bézier control points
⇒ Bézier controls polygon
— not assumed to be closed
Parametric Curves

To create plots: evaluate the cubic curve at many $t$-parameters
— $t \in [0, 1]$
— Example: evaluate at 50 points

$$t = 0, 1/50, 2/50, \ldots, 49/50, 1$$

Evaluation points connected by straight line segments
Choose enough $t$-parameters so the curve looks smooth
⇒ plotting a discrete approximation of the curve

Two examples that differ in the location of $b_0$ only
Parametric Curves

Evaluation: the de Casteljau algorithm

Generate one point
— Pick a parameter value \( t \in [0, 1] \)
— Linear interpolation on each leg

\[
\begin{align*}
b_0^1(t) &= (1 - t)b_0 + tb_1 \\
b_1^1(t) &= (1 - t)b_1 + tb_2 \\
b_2^1(t) &= (1 - t)b_2 + tb_3
\end{align*}
\]

Repeat on new polygon

\[
\begin{align*}
b_0^2 &= (1 - t)b_0^1(t) + tb_1^1(t) \\
b_1^2 &= (1 - t)b_1^1(t) + tb_2^1(t)
\end{align*}
\]

Repeat

\[
b_0^3(t) = (1 - t)b_0^2(t) + tb_1^2(t)
\]

\( \Rightarrow \) Point on the Bézier curve
Points $b_i^j$ are called intermediate Bézier points.

Except for (input) Bézier polygon each point is a function of $t$. 
Example: de Casteljau algorithm at $t = 1/2$

\[
\begin{align*}
\mathbf{b}_0 &= \begin{bmatrix} 4 \\ 4 \end{bmatrix} & \mathbf{b}_1 &= \begin{bmatrix} 0 \\ 8 \end{bmatrix} & \mathbf{b}_2 &= \begin{bmatrix} 8 \\ 8 \end{bmatrix} & \mathbf{b}_3 &= \begin{bmatrix} 8 \\ 0 \end{bmatrix} \\
\mathbf{b}_0^1 &= \frac{1}{2} \mathbf{b}_0 + \frac{1}{2} \mathbf{b}_1 = \begin{bmatrix} 2 \\ 6 \end{bmatrix} \\
\mathbf{b}_1^1 &= \frac{1}{2} \mathbf{b}_1 + \frac{1}{2} \mathbf{b}_2 = \begin{bmatrix} 4 \\ 8 \end{bmatrix} \\
\mathbf{b}_2^1 &= \frac{1}{2} \mathbf{b}_2 + \frac{1}{2} \mathbf{b}_3 = \begin{bmatrix} 8 \\ 4 \end{bmatrix} \\
\mathbf{b}_0^2 &= \frac{1}{2} \mathbf{b}_0^1 + \frac{1}{2} \mathbf{b}_1^1 = \begin{bmatrix} 3 \\ 7 \end{bmatrix} \\
\mathbf{b}_1^2 &= \frac{1}{2} \mathbf{b}_1^1 + \frac{1}{2} \mathbf{b}_2^1 = \begin{bmatrix} 6 \\ 6 \end{bmatrix} \\
\mathbf{b}_0^3 &= \frac{1}{2} \mathbf{b}_0^2 + \frac{1}{2} \mathbf{b}_1^2 = \begin{bmatrix} \frac{9}{2} \\ \frac{13}{2} \end{bmatrix}
\end{align*}
\]
Parametric Curves

Equation for a cubic Bézier curve

Expand: \( \mathbf{b}_0^3(t) = (1 - t)\mathbf{b}_0^2(t) + t\mathbf{b}_1^2(t) \)

\[
\mathbf{b}_0^3 = (1 - t)\mathbf{b}_0^2 + t\mathbf{b}_1^2 \\
= (1 - t) \left[(1 - t)\mathbf{b}_0^1 + t\mathbf{b}_1^1\right] + t \left[(1 - t)\mathbf{b}_1^1 + t\mathbf{b}_2^1\right] \\
= (1 - t) \left[(1 - t)\mathbf{b}_0 + t\mathbf{b}_1\right] + t \left[(1 - t)\mathbf{b}_1 + t\mathbf{b}_2\right] \\
+ t \left[(1 - t)\mathbf{b}_1 + t\mathbf{b}_2\right] + t \left[(1 - t)\mathbf{b}_2 + t\mathbf{b}_3\right]
\]

Collect terms with the same \( \mathbf{b}_i \)

\[
\mathbf{b}_0^3(t) = (1 - t)^3\mathbf{b}_0 + 3(1 - t)^2t\mathbf{b}_1 + 3(1 - t)t^2\mathbf{b}_2 + t^3\mathbf{b}_3
\]

General form of a cubic Bézier curve
\( \mathbf{b}_0^3(t) \) traces out a curve as \( t \) traces out values between 0 and 1
Shorter notation: \( \mathbf{b}(t) \) instead of \( \mathbf{b}_0^3(t) \)
Parametric Curves

Bernstein basis functions

Cubic polynomials — degree 3

\[ B_0^3(t) = (1 - t)^3 \]
\[ B_1^3(t) = 3(1 - t)^2 t \]
\[ B_2^3(t) = 3(1 - t)t^2 \]
\[ B_3^3(t) = t^3 \]

All cubic polynomials can be expressed in terms of Bernstein polynomials

\[ b_0^3(t) = B_0^3(t)b_0 + B_1^3(t)b_1 + B_2^3(t)b_2 + B_3^3(t)b_3 \]

⇒ form a basis for all cubic
A subset of intermediate Bézier points form two cubic polygons — Mimic the curve’s shape
Curve segment from $\mathbf{b}_0$ to $\mathbf{b}_0^3(t)$
$$\mathbf{b}_0, \mathbf{b}_0^1, \mathbf{b}_0^2, \mathbf{b}_0^3$$
Curve segment from $\mathbf{b}_0^3(t)$ to $\mathbf{b}_3$
$$\mathbf{b}_0^3, \mathbf{b}_1^2, \mathbf{b}_2^1, \mathbf{b}_3$$
Identify polygons in schematic
$$\mathbf{b}_0, \mathbf{b}_1^1, \mathbf{b}_2^1, \mathbf{b}_3^1, \mathbf{b}_0$$
Properties of Bézier Curves

Endpoint interpolation: \( b(0) = b_0 \) and \( b(1) = b_3 \)

Affine invariance: affine map of the control polygon then curve undergoes the same transformation

\[ (1 - t)^3 + 3(1 - t)^2 t + 3(1 - t)t^2 + t^3 = [(1 - t) + t]^3 = 1 \]

Every point on the curve is a *barycentric combination* of the control points.
Properties of Bézier Curves

Bernstein polynomials are nonnegative for $t \in [0, 1]$

⇒ Every point on the curve is a convex combination of the control points

⇒ Convex hull property: for $t \in [0, 1]$ the curve lies in the convex hull of the control polygon

Extrapolation: evaluate the curve for $t$-values outside of $[0, 1]$

— Can no longer predict the shape of the curve

The convex hull lies inside the minmax box
The Matrix Form

Bézier curve expressed using dot products:

\[ b(t) = \begin{bmatrix} b_0 & b_1 & b_2 & b_3 \end{bmatrix} \begin{bmatrix} (1 - t)^3 \\ 3(1 - t)^2 t \\ 3(1 - t)t^2 \\ t^3 \end{bmatrix} \]

Most well-known: cubic polynomials as combinations of the monomials:

\[ 1, t, t^2, t^3 \]

Rewrite the Bézier curve:

\[ b(t) = b_0 + 3t(b_1 - b_0) + 3t^2(b_2 - 2b_1 + b_0) + t^3(b_3 - 3b_2 + 3b_1 - b_0) \]

⇒ Matrix form of a Bézier curve

\[ b(t) = \begin{bmatrix} b_0 & b_1 & b_2 & b_3 \end{bmatrix} \begin{bmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ t \\ t^2 \\ t^3 \end{bmatrix} \]
A curve in monomial form:

\[ b(t) = b_0 + 3t(b_1 - b_0) + 3t^2(b_2 - 2b_1 + b_0) + t^3(b_3 - 3b_2 + 3b_1 - b_0) \]

\[ = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \]

Geometrically: \( a_0 = b_0 \) is a point \( a_1, a_2, a_3 \) are vectors

Using the dot product form:

\[ b(t) = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 \end{bmatrix} \begin{bmatrix} 1 \\ t \\ t^2 \\ t^3 \end{bmatrix} \]

\( \Rightarrow \) the monomial \( a_i \) are defined as

\[ \begin{bmatrix} a_0 & a_1 & a_2 & a_3 \end{bmatrix} = \begin{bmatrix} b_0 & b_1 & b_2 & b_3 \end{bmatrix} \begin{bmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \]
The inverse process:
Given a curve in monomial form, how to write it as a Bézier curve?

\[
\begin{bmatrix}
  b_0 & b_1 & b_2 & b_3
\end{bmatrix} = \begin{bmatrix}
  a_0 & a_1 & a_2 & a_3
\end{bmatrix}
\begin{bmatrix}
  1 & -3 & 3 & -1 \\
  0 & 3 & -6 & 3 \\
  0 & 0 & 3 & -3 \\
  0 & 0 & 0 & 1
\end{bmatrix}^{-1}
\]

A matrix inversion is all that is needed here!
— Matrix is nonsingular
⇒ Any cubic curve can be written in either Bézier or monomial form
Cubic Bézier curve: two (in 2D) or three (in 3D) cubic equations in $t$

$$b^3_0(t) = (1 - t)^3b_0 + 3(1 - t)^2tb_1 + 3(1 - t)t^2b_2 + t^3b_3$$

First derivative vector: derivative in each of the components

$$\frac{db(t)}{dt} = -3(1 - t)^2b_0 - 6(1 - t)tb_1 + 3(1 - t)^2b_1$$
$$- 3t^2b_2 + 6(1 - t)t^2b_2 + 3t^2b_3$$

Rearrange and use abbreviation $\frac{db(t)}{dt} = \dot{b}(t)$

$$\dot{b}(t) = 3(1 - t)^2[b_1 - b_0] + 6(1 - t)t[b_2 - b_1] + 3t^2[b_3 - b_2]$$

⇒ Derivative curve is degree two
⇒ Derivative curve has control vectors
Derivatives

One very nice feature of the de Casteljau algorithm: intermediate Bézier points lead to simple expression of the derivative

\[ \dot{\mathbf{b}}(t) = 3 \left[ \mathbf{b}_2(t) - \mathbf{b}_0(t) \right] \]

Very simple form at the endpoints

\[ \dot{\mathbf{b}}(0) = 3[\mathbf{b}_1 - \mathbf{b}_0] \quad \dot{\mathbf{b}}(1) = 3[\mathbf{b}_3 - \mathbf{b}_2] \]

Control polygon is tangent to the curve at the curve’s endpoints
Derivatives

Example: Compute first derivative vector for \( t = 1/2 \)

Evaluate the direct equation

\[
\dot{b} \left( \frac{1}{2} \right) = 3 \cdot \frac{1}{4} \left[ \begin{bmatrix} 0 \\ 8 \end{bmatrix} - \begin{bmatrix} 4 \\ 4 \end{bmatrix} \right] + 6 \cdot \frac{1}{4} \left[ \begin{bmatrix} 8 \\ 8 \end{bmatrix} - \begin{bmatrix} 0 \\ 8 \end{bmatrix} \right] + 3 \cdot \frac{1}{4} \left[ \begin{bmatrix} 8 \\ 0 \end{bmatrix} - \begin{bmatrix} 8 \\ 8 \end{bmatrix} \right] = \begin{bmatrix} 9 \\ -3 \end{bmatrix}
\]

Use intermediate control points

\[
\dot{b} \left( \frac{1}{2} \right) = 3 \left[ b_1^2 \left( \frac{1}{2} \right) - b_0^2 \left( \frac{1}{2} \right) \right] = 3 \left[ \begin{bmatrix} 6 \\ 6 \end{bmatrix} - \begin{bmatrix} 3 \\ 7 \end{bmatrix} \right] = \begin{bmatrix} 9 \\ -3 \end{bmatrix}
\]
Derivatives

The derivative of a curve is a \textit{vector}
— It is tangent to the curve

Can interpret it as a \textit{velocity vector}
— Interpret parameter $t$ as time
  then arrive at $\mathbf{b}(t)$ at time $t$ with velocity $\mathbf{\dot{b}}(t)$
— Large magnitude of the tangent vector $\Rightarrow$ moving fast

If we rotate the control polygon, the curve will follow
— And so will all of its derivative vectors

In calculus: a “horizontal tangent” has a special meaning
— It indicates an extreme value of a function
— Notion of an extreme value is meaningless for parametric curves
  ◊ Term “horizontal tangent” depends on the curve’s orientation
  ◊ Not a property of the curve itself
Derivatives

Second derivative vector: may interpret as acceleration

\[ \dot{\mathbf{b}}(t) = 3(1 - t)^2[\mathbf{b}_1 - \mathbf{b}_0] + 6(1 - t)t[\mathbf{b}_2 - \mathbf{b}_1] + 3t^2[\mathbf{b}_3 - \mathbf{b}_2] \]
\[ \ddot{\mathbf{b}}(t) = -6(1 - t)[\mathbf{b}_1 - \mathbf{b}_0] - 6t[\mathbf{b}_2 - \mathbf{b}_1] + 6(1 - t)[\mathbf{b}_2 - \mathbf{b}_1] + 6t[\mathbf{b}_3 - \mathbf{b}_2] \]
\[ = 6(1 - t)[\mathbf{b}_2 - 2\mathbf{b}_1 + \mathbf{b}_0] + 6t[\mathbf{b}_3 - 2\mathbf{b}_2 + \mathbf{b}_1] \]

Via the de Casteljau algorithm:

\[ \ddot{\mathbf{b}}(t) = 6 \left[ \mathbf{b}^{1}_2(t) - 2\mathbf{b}^{1}_1(t) + \mathbf{b}^{1}_0(t) \right] \]

Simple form at endpoints: \[ \ddot{\mathbf{b}}(0) = 6[\mathbf{b}_2 - 2\mathbf{b}_1 + \mathbf{b}_0] \]
Composite Curves

For “real” shapes need to use many cubic Bézier curves
— Smooth overall curve ⇒ pieces must join smoothly

The letter “D” as a collection of cubic Bézier curves
— Only one Bézier polygon of many is shown
Composite Curves

Smoothly joining Bézier curves

Two Bézier curves: $b_0, b_1, b_2, b_3$ and $c_0, c_1, c_2, c_3$

Common point $b_3 = c_0$

Same tangent vector direction:

$$c_1 - c_0 = c[b_3 - b_2]$$

For some positive real number $c$

$\Rightarrow b_2, b_3 = c_0, c_1$ collinear
Imagine driving with constant speed. Curviness of road proportional to turning of steering wheel.

Sample tangents at various points:
- Successive tangents differ significantly
  ⇒ curvature is high
- Successive tangents are almost identical
  ⇒ curvature is low

**Curvature:** rate of change of tangents
The Geometry of Planar Curves

Tangent determined by the first derivative
⇒ its rate of change should also involve the second derivative

Curvature: \( \kappa(t) = \frac{\| \dot{b} \wedge \ddot{b} \|}{\| \dot{b} \|^3} \)

2D: can use special formula

\( \kappa(t) = \frac{\| \dot{b} \dddot{b} \|}{\| \dot{b} \|^3} \)

⇒ 2D curvature is \textit{signed}
The Geometry of Planar Curves

**Inflection point:** a point where $\kappa = 0$

— Curvature changes sign on either side of the inflection point
— First and second derivative vectors are parallel or linearly dependent
Moving along a Curve

Curve motions: “B” sliding along a Bézier curve
Moving along a Curve

Position an object at a point on a 2D curve

Point on curve: $b(t)$

Tangent $\dot{b}$

Vector $n$ perpendicular to tangent

$\Rightarrow$ Local coordinate system
Moving along a Curve

Position an object at a point on a 3D curve

Frenet frame

\[ f_1 = \frac{\dot{b}(t)}{\|\dot{b}(t)\|} \]

\[ f_3 = \frac{\dot{b}(t) \wedge \ddot{b}(t)}{\|\dot{b}(t) \wedge \ddot{b}(t)\|} \]

\[ f_2 = f_3 \wedge f_1 \]

Osculating plane: defined by \( \dot{b}(t) \) and \( \ddot{b}(t) \)
Moving along a Curve

Suppose an object is in some local coordinate system with axes $u_1, u_2, u_3$

Any point of the object has coordinates $u$

Mapped to: $x(t, u) = b(t) + u_1f_1 + u_2f_2 + u_3f_3$

A typical application: robot motion

— A robot arm is moved along a curve
- linear interpolation
- parametric curve
- de Casteljau algorithm
- cubic Bézier curve
- subdivision
- affine invariance
- convex hull property
- Bernstein polynomials
- basis function
- barycentric combination
- matrix form
- cubic monomial curve

- Bernstein and monomial conversion
- nonsingular
- first derivative
- second derivative
- parallelogram rule
- composite Bézier curves
- curvature
- inflection point
- Frenet frame
- osculating plane