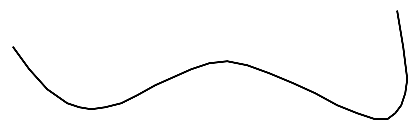
CS181a: Computer Animation

Curves

Z Sweedyk

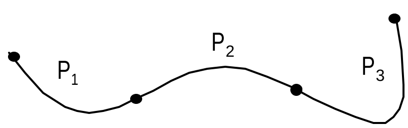
Curve



How should we represent a curve?

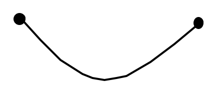
- Efficiency
- Usability

Complicated Curves



Simple curves connected end-to-end

Simple Curves



How should we represent a simple curve?

- Efficiency
- Usability
- Boundary constraints

Outline

- Representation
- Some Examples: Hermitian, Bezier
- Computation: Recursive subdivision

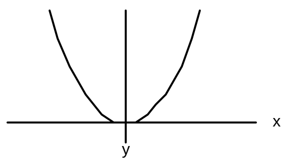
Curve Representation

- Explicit
- Implicit
- Parametric

Explicit

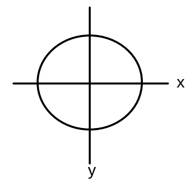
Curve is the trace of a function

Example: $y=x^2/4$



Explicit: problem

Many useful curves cannot be represented by explicit functions



Implicit

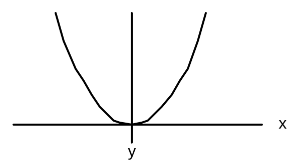
Curve is the zero Loci of a function

Example: $f(x,y) = 4y-x^2$

Implicit

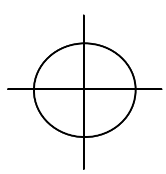
Curve is the zero Loci of a function

Example: $f(x,y) = 4y-x^2$



Implicit: more flexibility

$F(x,y)=x^2+y^2-r^2$



Implicit: problems

- How can we find the zero loci of an function f(x,y)?
- How can we express a half-circle?
- How can we specify boundary conditions for joins?

Parametric

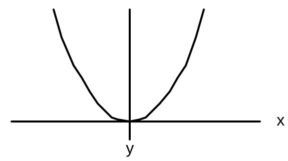
Curve is the range of a function

Example: $x = 2t$, $y = t^2$

Parametric

Curve is the range of a function

Example: $x = 2t$, $y = t^2$



Parametric: tradeoffs

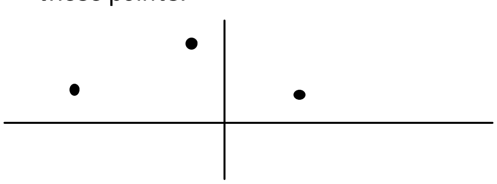
- Plus:
  - Computationally easy to find points on curve
  - Easy to specify portions of curves
  - Easy to specify boundary conditions
- Minus:
  - Without modeling tool the specification is not intuitive

Parametric cubic polynomials

- Polynomials are expressive and can be efficiently computed
- Lower degree polynomials can't express non-planar curves
- Higher degree polynomials
  - Wiggle
  - Computationally more expensive

Interpolating polynomials

- Give me a polynomial curve through these points:

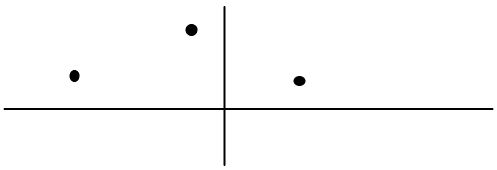


Interpolating polynomials

- There is a unique polynomial curve of degree 2 through these points.
- $Y= y_0(x-x_1)(x-x_2)/(x_0-x_1)(x_0-x_2)$ +
  $y_1(x-x_0)(x-x_2)/(x_1-x_0)(x_1-x_2)$ +
  $y_2(x-x_0)(x-x_1)/(x_2-x_0)(x_2-x_1)$ +

What about boundary conditions?

- Give me a polynomial curve through these points:

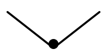


Parametric Continuity

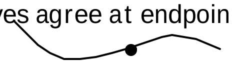
C^i: The 0^{th}, 1^{st}, 2^{nd}, …, i^{th} derivative of adjacent curves agree at their connecting endpoints.

Parametric Continuity

- C^0: adjacent curves connect at endpoints



- C^1: the 1^{st} derivative of adjacent curves agree at endpoints

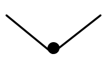


Geometric Continuity

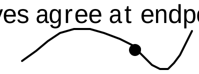
G^i: The 0^{th}, 1^{st}, 2^{nd}, …, i^{th} derivative of adjacent curves are proportional at endpoints

Geometric Continuity

- G^0: adjacent curves connect at endpoints



- G^1: the 1^{st} derivative of adjacent curves agree at endpoints



Hermitian

- Specify endpoint position
- Specify endpoint tangent

Hermitian

- $X(t) = at^3 + bt^2 + ct + d$
- $X(0) = 3$, $X(1) = 2$
- $X'(0) = 1$, $X'(1) = 0$
- Write 4 equations that determine the coefficients a, b, c, and d.

Hermitian: Constraints

- X(0):
- X(1):
- X'(0):
- X'(1):

Hermitian: Constraints

- X(0): $d = 3$
- X(1): $a+b+c+d = 2$
- X'(0): $c = 1$
- X'(1): $3a+2b+c=0$

Hermitian Matrix Form

-	-	-	-	a		-
-	-	-	-	b	=	-
-	-	-	-	c		-
-	-	-	-	d		-

Hermitian Matrix Form

0	0	0	1	a		3
1	1	1	1	b	=	2
0	0	1	0	c		1
3	2	1	0	d		0

Find X(t)

Solve for a, b, c, d: Hint

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Hermitian Matrix: X

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ -5 \\ 1 \\ 3 \end{bmatrix}$$

Parametric equation: X

$$X(t) = 3t^3 - 5t^2 + t + 3$$

General Matrix: X

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X(0) \\ X(1) \\ X'(0) \\ X'(1) \end{bmatrix}$$

Hermitian Matrix: Y

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Y(0) \\ Y(1) \\ Y'(0) \\ Y'(1) \end{bmatrix}$$

Hermitian Basis Matrix

$$\begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Example

- X(0) = 3, X(1) = 2, X'(0) = 1, X'(1) = 0
- Y(0) = 2, Y(1) = 2, X'(0) = 0, X'(1) = 1

- Write the equations
- Plot the curve for t in [0,1]

Equations

- X(0) = 3, X(1) = 2, X'(0) = 1, X'(1) = 0
- Y(0) = 2, Y(1) = 2, X'(0) = 0, X'(1) = 1

- Equations:
- $X(t) = 3t^3 - 5t^2 + t + 3$
- $Y(t) = t^3 - t^2 + 2$

Plot

- X(0) = 3, X(1) = 2, X'(0) = 1, X'(1) = 0
- Y(0) = 2, Y(1) = 2, X'(0) = 0, X'(1) = 1

Hermitian Description

- Endpoint constraints
- Basis Matrix
- Basis (blending) Functions

Hermitian: X(t)

$$X(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

General Matrix: X

$$X(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X(0) \\ X(1) \\ X'(0) \\ X'(1) \end{bmatrix}$$

Blending Functions: X

$$X(t) = \begin{bmatrix} P_1(t) & P_2(t) & P_3(t) & P_4(t) \end{bmatrix} \begin{bmatrix} X(0) \\ X(1) \\ X'(0) \\ X'(1) \end{bmatrix}$$

Hermitian Blending Functions

$$\begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} =$$

$$\begin{bmatrix} 2t^3-3t^2+1, & -2t^3+3t^2, & t^3-2t^2+t, & t^3-t^2 \end{bmatrix}$$

$P_1(t)$ $P_2(t)$ $P_3(t)$ $P_4(t)$

Plot the Hermitian Blending Functions

Hermitian: problem

1. Specifying derivatives is not intuitive for users.
2. Not invariant under affine transformations

Properties of Cubic Bezier Curves

- Control points p_0, p_1, p_2, p_3
- Curve starts at p_0 and ends at p_3.
- Line segments p_0-p_1 and p_3-p_2 are tangent to the curve at, respectively, p_0 and p_3.
- The curve lies within the convex hull of the control points.
- Curve is invariant under affine transformations.

Lab

- Download bezier.cpp from /cs/cs155/labs
- Compile and Play
  - Right click to move red point
  - Left click to select new red point
  - Type “a” to then right click (3 times) to add new control points
  - Type “d” to delete last point

Cubic Bezier Matrix: X

$$X(t) = \begin{bmatrix} -1 & -3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & -3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

x_i is the x-coordinate of the i-th control point

Cubic Bezier Blending Functions

(Bernstein polynomials)

$x(t) = x_0 b_0(t) + x_1 b_1(t) + x_2 b_2(t) + x_3 b_3(t)$

- x_i is x-coordinate of the i^{th} control point
- $b_0(t) = (1-t)^3$
- $b_1(t) = 3t(1-t)^2$
- $b_2(t) = 3t^2(1-t)$
- $b_3(t) = t^3$

Computation of points on curve

- Direct computation
  - inefficient
- Forward differencing
  - error-prone
- Recursive, adaptive subdivision: deCasteljau algorithm

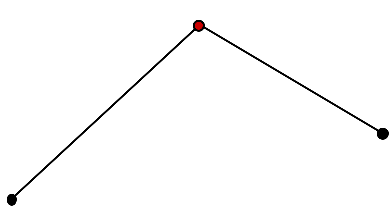
Quadratic Bezier Blending Functions

(Bernstein polynomials)

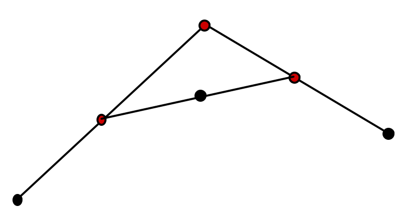
$x(t) = x_0 b_{2,0}(t) + x_1 b_{2,1}(t) + x_2 b_{2,2}(t)$

- x_i is x-coordinate of the i^{th} control point
- $b_{2,0}(t) = (1-t)^2$
- $b_{2,1}(t) = 2t(1-t)$
- $b_{2,2}(t) = t^2$

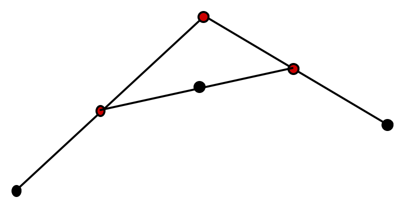
Quadratic Bezier I



Quadratic Bezier: II



Compute Quadratic Bezier: IV



1/30/02 CS181a Lec 02 55

Why does this work?

- To be continued…
  - Subdivision Curves and Surfaces

1/30/02 CS181a Lec 02 56