Lecture 3

178 359
Simulation and Modeling

Pattarawit Polpinit

Department of Computer Engineering
Khon Kaen Uiversity

Overview

Curve Modeling

- Bezier Curve
  - Parametric representation
  - Properties
  - Matrix representation
- de Calteljau Algorithm
  - The algorithm
  - Its complexity
- Related Derivative of the Curve

Bezier Curve: History

- Bezier curves were widely publicized by the french engineer Pierre Bezier, who used them to design automobile bodies.
- However, the curves were first invented by Paul de Casteljau in 1959.
- de Casteljau's algorithm, a numerically stable method was used to evaluate Bezier curves.
- Bezier curves are extensively used in computer graphics and animations.

Bezier Curve: Definition

- Given $n+1$ points, $p_1, p_2, \ldots, p_n$ in $\mathbb{R}^3$.
- A Bezier curve of degree n (order $n+1$) defined by these points, can be expressed as

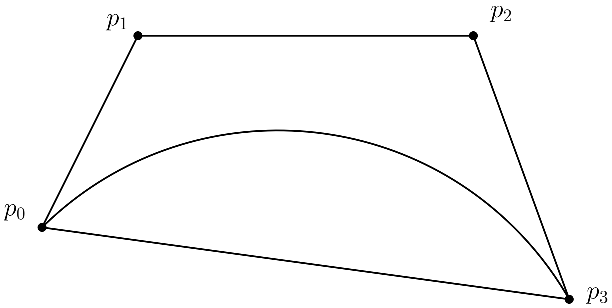
$$\mathbf{B}(t) = p_0 B_0^n(t) + p_1 B_1^n(t) + \ldots + p_n B_n^n(t)$$

where

$$B_i^n(t) = \binom{n}{i} \cdot t^i \cdot (1-t)^{n-i} = \frac{n!}{i!(n-i)!} \cdot t^i \cdot (1-t)^{n-i}$$

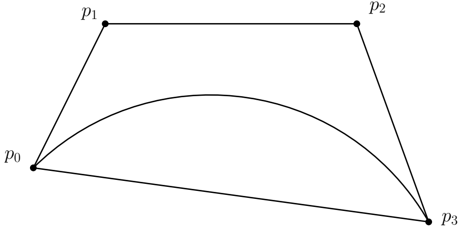
is a Bernstein polynomials and $\binom{n}{i}$ is a binomial coefficients.

Cubic Bezier Curve



Cubic Bezier Curve: Description

- Four points p_0, p_1, p_2 and p_3 are needed in the plane to define a cubic Bezier curve.
- The curve starts at p_0 going toward p_1 and arrives at p_3 coming from the direction of p_2.
- Usually, it will not pass through p_1 or p_2; these points are only there to provide directional information.
- The distance between p_0 and p_1 determines"how long" the curve moves into direction p_2 before turning towards p_3.



Cubic Bezier Curve: The Equation

$$\mathbf{B}(t) = \sum_{i=0}^{3} p_i B_i^3(t) = p_0 B_0^3(t) + p_1 B_1^3(t) + p_2 B_2^3(t) + p_3 B_3^3(t)$$

where

$$\begin{aligned}
B_0^3(t) &= \binom{3}{0} \cdot t^0 \cdot (1-t)^{3-0} && = (1-t)^3 && \geq 0 \\
B_1^3(t) &= \binom{3}{1} \cdot t^1 \cdot (1-t)^{3-1} && = 3t(1-t)^2 && \geq 0 \\
B_2^3(t) &= \binom{3}{2} \cdot t^2 \cdot (1-t)^{3-2} && = 3t^2(1-t) && \geq 0 \\
B_3^3(t) &= \binom{3}{3} \cdot t^3 \cdot (1-t)^{3-3} && = t^3 && \geq 0
\end{aligned}$$

Convex Combination

Cubic Bezier Curve is a convex combination.

Proof: The curve is a convex combination iff it is a linear combination, affine combination and all coefficients are nonnegative.

- It is obvious that $\mathbf{B}(t)$ is a linear combination.
- Affine combination:

$$\sum_{i=0}^{3} B_i^3(t)$$

$$\begin{aligned}&= (1-t)^3 + 3t(1-t)^2 + 3t^2(1-t) + t^3 \\ &= (1-3t+3t^2-t^3) + (3t-6t+3t^2) + (3t^2-3t^3) + t^3 \\ &= 1\end{aligned}$$

- $B_i^3(t) \geq 0 \text{ for } i = \{0, 1, 2, 3\}$ as shown previously.

Matrix Representation

$$\mathbf{B}(t) = G \cdot M \cdot T$$

$$\begin{aligned}\mathbf{B}(t) = & \left(-t^3 + 3t^2 - 3t + 1\right) \cdot p_0 + \left(3t^3 - 6t^2 + 3t\right) \cdot p_1 \\ & + \left(-3t^3 + 3t^2\right) \cdot p_2 + \left(t^3\right) \cdot p_3\end{aligned}$$

where

$$G = \begin{bmatrix} p_0 & p_1 & p_2 & p_3 \end{bmatrix}$$

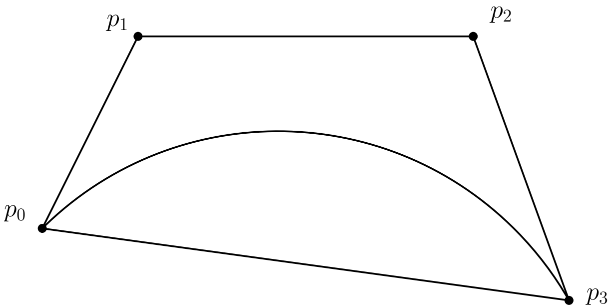
$$M = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$T = \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix}$$

Properties of Bezier Curve

1. The Bezier curve passes through (interpolates) two endpoints.

$$\mathbf{B}(0) = p_0 \text{ and } \mathbf{B}(1) = p_n$$



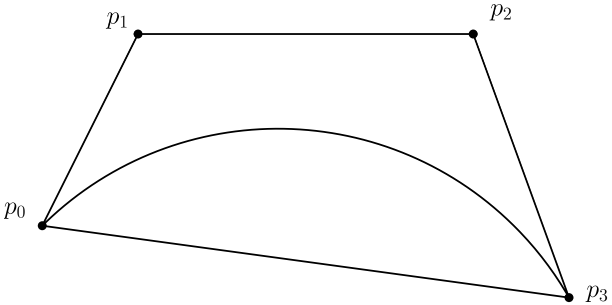
Properties of Bezier Curve (2)

2. Affince combinations

$$\sum_{i=0}^{n} B_i^n(t) = 1, \forall t \in [0, 1]$$

Properties of Bezier Curve (3)

3. Convex combinations: the Bezier curve lies inside the convex hull of the control net.



de Casteljau Algorithm

- named after Paul de Casteljau.
- A recursive method used to evaluate Bernstien polynomial of Bezier curves.
- Sometimes used to split a single Beizer curve into two.
- Slower than most direct approach, but numerically more stable.

So why de Casteljau Algorithm?

- Objective is to find a point on a Bezier curve.
- We can obviously plug in t then compute every Bernstien polynomials; their products and their corresponding control points.
- This work OK, but not numerically stable, namely could introduce numerically error.

de Casteljau Algorithm (2)

Recall that the Bezier curve is

$$\mathbf{B}(t) = \sum_{i=0}^{n} B_i^n(t) \cdot b_i$$

Note: hereafter we shall denote a point with b_i.

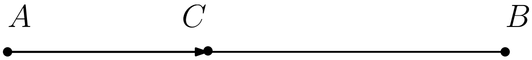
We can calculate the points on the curve corresponding to t by

$$\begin{aligned} b_i^0(t) &= b_i \\ b_i^r(t) &= (1-t) \cdot b_i^{r-1}(t) + t \cdot b_{i+1}^{r-1}(t) \end{aligned}$$

where $r = 1, \ldots, n$ and $i = 0, \ldots, n-r$.

Fundamental Concept

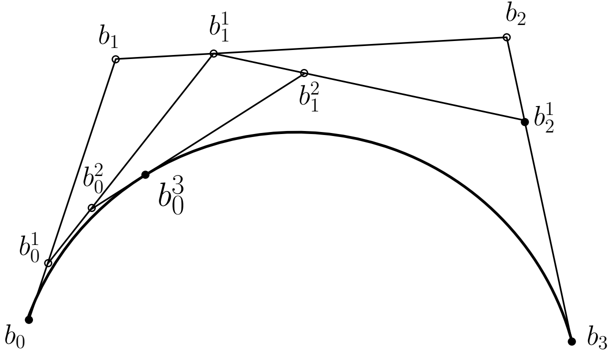
- The concept is, given t, we want to find a point C such that it divide a line segment AB into AC and CB with a ratio of t and $1 - t$.



- C can be found by $(1 - t)A + t \cdot B$
- Hence, to find a point of degree n at t, we divide a line segment of a line segment constructed from points of degree $n - 1$ at t.

de Casteljau Algorithm: Example

Example: Find a point on a Bezier curve where $t = 1/4$.



Graphically representing a cubic Bezier curve (degree 3) and the calculated point where $t = 1/4$.

de Casteljau Algorithm: Example

Example: Calculate a point on the cubic Bezier curve when $t = 1/2$ using de Casteljau algorithm

Answer: A point on the curve when $t = 1/2$ is $b_0^3(1/2)$. Graphically, that is

de Casteljau Algorithm: Example (cont.)

Answer: Expression can be computed by,

Complexity of de Casteljau Algorithm

Using de Casteljau algorithm

$$
\begin{aligned}
b_i^0(t) &= b_i \\
b_i^r(t) &= (1-t)\cdot b_i^{r-1} + t\cdot b_{i+1}^{r-1}(t)
\end{aligned}
$$

where $r = 1, \ldots, n$ and $i = 0, \ldots, n-r$.

- There are 1 addition (1A) and 2 multiplications (2M) in each recursion.
- Thus, the complexity can be calculated by

$$\sum_{r=1}^{n}\sum_{i=0}^{n+r}(A+2M)$$

Complexity (2)

$$
\begin{aligned}
\sum_{r=1}^{n}\sum_{i=0}^{n-r}(A+2M) &= \sum_{r=1}^{n}(n-r+1)(A+2M) \\
&= \sum_{r=1}^{n}(n-r+1)(A) + \sum_{r=1}^{n}(n-r+1)(2M) \\
&= \left[\sum_{r=1}^{n}(n+1)A - \sum_{r=1}^{n} rA\right] \\
&\quad + \left[\sum_{r=1}^{n}(n+1)2M - \sum_{r=1}^{n}(r)2M\right] \\
&= \left[n(n+1)A - \frac{n(n+1)}{2}A\right] \\
&\quad + [2n(n+1)M - n(n+1)M]
\end{aligned}
$$

Complexity (3)

$$\begin{aligned}
\sum_{r=1}^{n}\sum_{i=0}^{n-r}(A+2M) &= \sum_{r=1}^{n}(n-r+1)(A+2M) \\
&= \vdots \\
&= \left[n(n+1)A - \frac{n(n+1)}{2}A\right] \\
&\qquad + [2n(n+1)M - n(n+1)M] \\
&= \frac{n(n+1)}{2}A + n(n+1)M
\end{aligned}$$

- de Casteljau algorithm requires $n(n+1)/2$ additions and $n(n+1)$ multiplications.
- The time complexity is $O(n^2)$.