Prof. Dr. Leif Kobbelt
Patrick Schmidt, Joe Jakobi

Basic Techniques in Computer Graphics

Assignment 12

Date Published: January 24th 2023, Date Due: January 31st 2023

- All assignments (programming and theory) have to be completed in teams of 3–4 students. Teams with fewer than 3 or more than 4 students will receive no points.
- Hand in **one solution per team per assignment**.
- Every team must work independently. Teams with identical solutions will receive no points.
- Solutions are due on January 31st 2023 via Moodle. Late submissions will receive zero points. No exceptions!
- Instructions for **programming assignments**:
  - Make sure you are part of a Moodle group with 3-4 members. See "Group Management" in the Moodle course room.
  - Download the solution template (a zip archive) through the Moodle course room.
  - Unzip the archive and populate the `assignmentXX/MEMBERS.txt` file. The names and student ids listed in this file **must match** your moodle group **exactly**.
  - Complete the solution.
  - Prepare a new zip archive containing your solution. It must contain exactly the files that you changed. **Only change the files you are explicitly asked to change in the task description.** The directory layout must be the same as in the archive you downloaded. (At the very least it must contain the `assignmentXX/MEMBERS.txt`.)
  - One team member uploads the zip archive through Moodle before the deadline, using the group submission feature.
  - Your solution must compile and run correctly **on our lab computers** by only inserting your **assignment.cc** and **shader files** into the Project. If it does not compile on our machines, you will receive no points. If in doubt you can test compilation in the virtual machine provided on our website.
- Instructions for **text assignments**:
  - Prepare your solution as a single pdf file per group. Submissions on paper will not be accepted.
  - If you write your solution by hand, write neatly! Anything we cannot decipher will receive zero points. No exceptions!
  - Add the names and student ID numbers of all team members to every pdf.
  - Unless explicitly asked otherwise, always justify your answer.
  - Be concise!
  - Submit your solution via Moodle, together with your coding submission.

Exercise 1 Bernstein Polynomials and Bézier Curves **[24 Points]**

(a) Derivatives of Bernstein Polynomials **[8 Points]**

Show: The first derivative of a Bernstein polynomial of degree n can be expressed as a difference of two Bernstein polynomials of degree $n-1$:

$$\frac{\mathrm{d}}{\mathrm{d}t} B_i^n(t) = n\left(B_{i-1}^{n-1}(t) - B_i^{n-1}(t)\right).$$

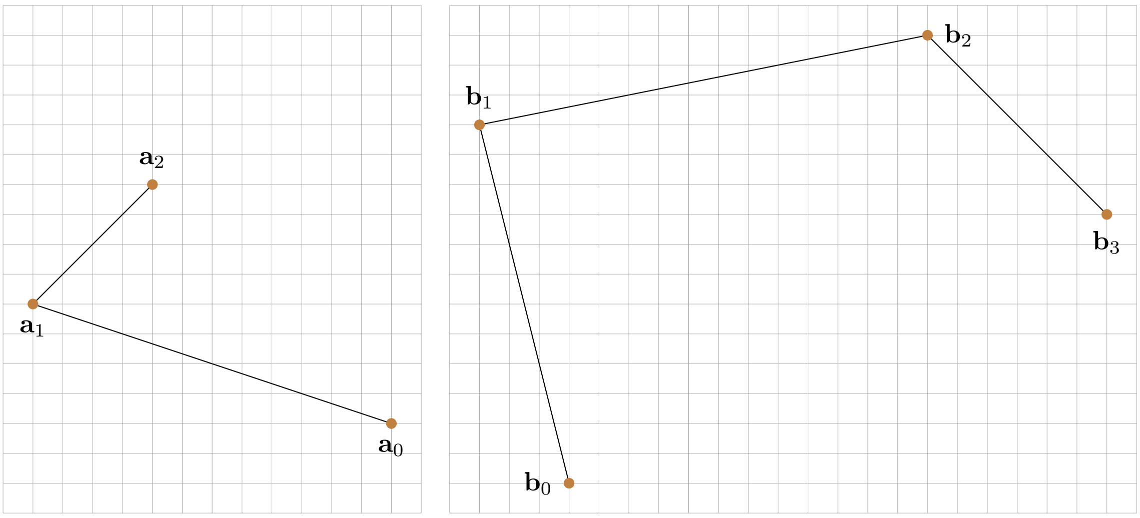
(b) Derivatives of Bézier Curves **[8 Points]**

Given a Bézier curve $\mathbf{b}(t)$ of degree n with control points $\mathbf{p}_0, \ldots, \mathbf{p}_n$, show that the derivative $\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{b}(t)$ is again a Bézier curve $\mathbf{b}'(t)$ (of degree $n-1$). What are the control points of the derivative curve?

Hint: Use the identity from task (a).

(c) De Casteljau Algorithm **[8 Points]**

Use the de Casteljau algorithm to evaluate the two Bézier curves defined by their control polygons below. Evaluate the left curve ($\mathbf{a}(t)$) at $t = \frac{1}{2}$ and the right curve ($\mathbf{b}(t)$) at $t = \frac{1}{3}$. Perform the evaluation graphically. You do not need to perform any computations.



Prof. Dr. Leif Kobbelt
Patrick Schmidt, Joe Jakobi

Exercise 2 Splines **[16 Points]**

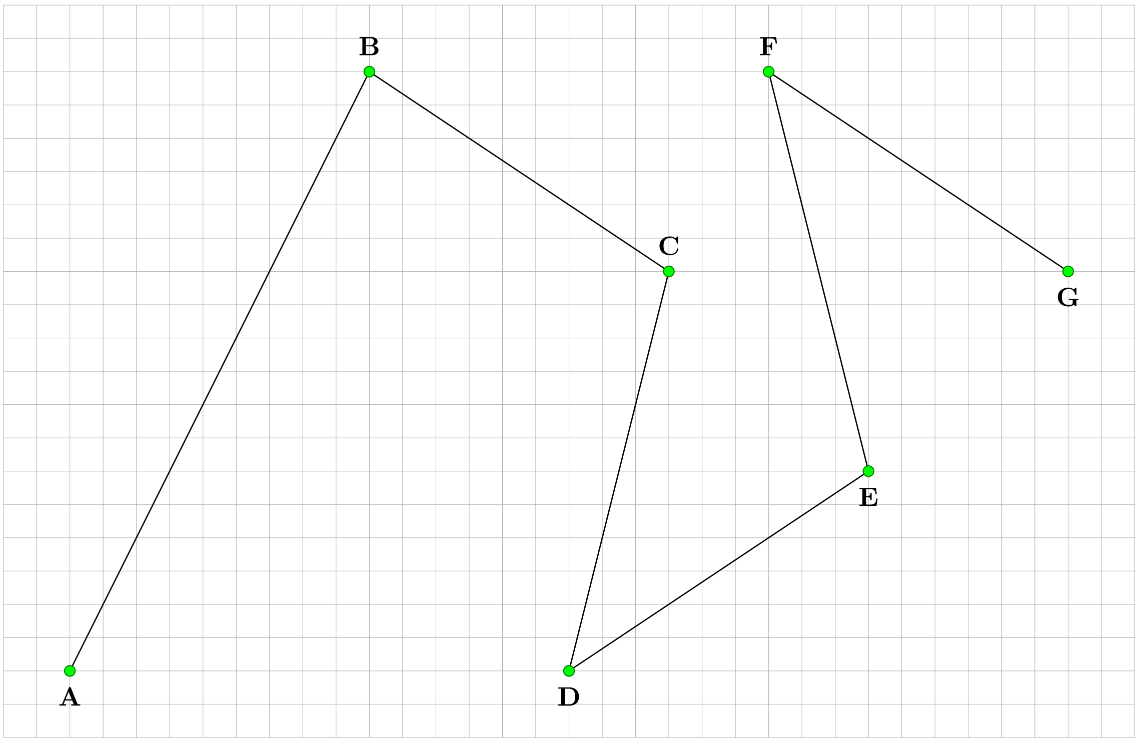
(a) B-Spline to Bézier Spline Conversion **[8 Point]**

In the figure below, you are given the control polygon $\mathbf{A}, \ldots, \mathbf{G}$ specifying a cubic B-spline curve. Using the A-frame construction method introduced in the lecture, graphically construct the control points for the Bézier spline describing the same curve. Your Bézier spline should consist of 4 segments, each of them a cubic (degree 3) Bézier curve:

1. $\mathbf{b}(t)$, close to the line segment $\overline{\mathbf{BC}}$, defined by control points $\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$
2. $\mathbf{c}(t)$, close to the line segment $\overline{\mathbf{CD}}$, defined by control points $\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$
3. $\mathbf{d}(t)$, close to the line segment $\overline{\mathbf{DE}}$, defined by control points $\mathbf{d}_0, \mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3$
4. $\mathbf{e}(t)$, close to the line segment $\overline{\mathbf{EF}}$, defined by control points $\mathbf{e}_0, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$

Include all auxiliary constructions in your solution. Also, don't forget to label all Bézier control points.

Note: You do not need to perform any computations. Do all constructions graphically using the methods presented in the lecture!



(b) Interpolating Bézier Spline Construction **[8 Point]**

Starting from an initial curve segment, incrementally construct a cubic (degree 3) Bézier spline with C^2 continuity that interpolates the points $\mathbf{B}, \mathbf{C}, \mathbf{D}$! The initial cubic Bézier curve $\mathbf{a}(t)$ is given by the control points $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ (indicated by the red control polygon).

1. First, construct the control polygon $\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$ of a cubic Bézier curve $\mathbf{b}(t)$ such that curves $\mathbf{a}$ and $\mathbf{b}$ join with C^2 continuity and its endpoint interpolates point $\mathbf{B}$ (i. e. $\mathbf{b}(1) = \mathbf{B}$).

2. Then, construct the control polygon $\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$ of a cubic Bézier curve $\mathbf{c}(t)$ such that curves $\mathbf{b}$ and $\mathbf{c}$ join with C^2 continuity and its endpoint interpolates point $\mathbf{C}$ (i. e. $\mathbf{c}(1) = \mathbf{C}$).

3. Finally, construct the control polygon $\mathbf{d}_0, \mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3$ of a cubic Bézier curve $\mathbf{d}(t)$ such that curves $\mathbf{c}$ and $\mathbf{d}$ join with C^2 continuity and its endpoint interpolates point $\mathbf{D}$ (i. e. $\mathbf{d}(1) = \mathbf{D}$).

Include all auxiliary constructions (such as A-frames) in your solution. Also, don't forget to label all Bézier control points.

Note: You do not need to perform any computations. Do all constructions graphically using the methods presented in the lecture!