MA 323 Geometric Modelling
Course Notes: Day 17
Bezier Splines I

David L. Finn

Today, we start considering piecewise Bezier curves or Bezier splines. We have previously introduced the idea of piecewise polynomial curves (Hermite curves), which was a direct generalization of piecewise linear curves (polylines) and piecewise circular curves. We now want to extend the idea of piecewise polynomial curves to Bezier curves, and introduce the idea of a composite Bezier curve or a *Bezier spline*. The word spline comes from naval architecture when knots, ropes and pulleys were used to bend the wooden planks that were used to build wooden ships. In fact, much of the terminology involved with splines comes from the physical constructions of naval architecture.

When constructing splines, it is typically done with respect to a smoothness condition. There are two types of smoothness that are considered, functional and geometric. These involve different notions of continuity. Functional continuity involves orders of continuity with respect to the parameter of the curve, while geometric continuity involves continuity with respect to the arc-length parameter of the curve. Today and tomorrow, we will only consider splines with respect to functional continuity. To look at geometric continuity, we need to look in more detail at the differential geometry of curves.

17.1 Piecewise Bezier Curves

We define a piecewise Bezier curve by giving two (or more) sets of control points. Let p_0^1, p_1^1, p_2^1, $\cdots$, p_n^1 be one set of control points and p_0^2, p_1^2, p_2^2, $\cdots$, p_n^2 be another set of control points. [It is convention that one joins curves of the same order, as using degree elevation we can always arrange the curves to have the same degree.] We can use each set to define a Bezier curve, let $b_1(t)$ be the curve with control points $\{p_i^1\}$ and let $b_2(t)$ be the curve with control points $\{p_j^2\}$. We can then define a piecewise Bezier curve by

$$b(s) = \begin{cases} b_1(s) & \text{if } 0 \le s \le 1 \\ b_2(s-1) & \text{if } 1 \le s \le 2 \end{cases}$$

We must define the curve in this manner so for each value of s we get back one point. We note that there is a slight problem with this definition. When $t = 1$, this curve is possibly multi-valued, depending on the values of the last control point of $b_1(t)$ and the first control point of $b_2(t)$, that is p_n^1 and p_0^2. The parameter t for each Bezier curve $b_i(t)$ is a *local parameter*, used to construct the Bezier curve, while the parameter s defining the piecewise Bezier curve is a *global parameter*.

Notice that a piecewise Bezier curve as defined above is continuous if and only if $p_n^1 = p_0^2$. This is due to the endpoint interpolation property for Bezier curves, and in this case the curve is single valued. When $p_n^1 \neq p_0^2$, a Bezier curve as defined above is not continuous and the definition is not single-valued. A piecewise Bezier curve that is continuous ($p_n^1 = p_0^2$) is said to have zeroth order continuity. Higher orders of functional continuity are defined by continuity of the derivatives of the curve at the joint point $p_n^1 = p_0^2$.

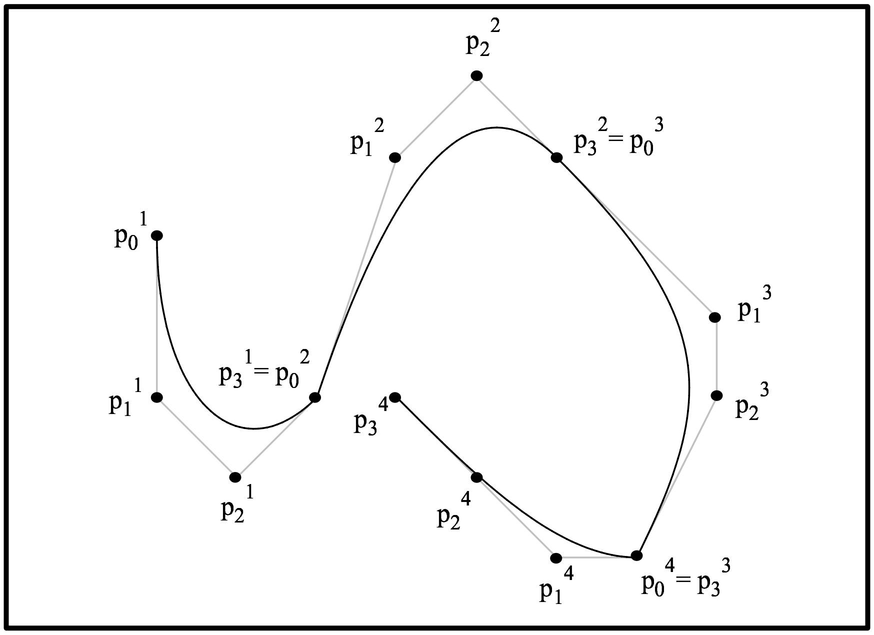

Figure 1: A piecewise Bezier curve consisting of 4 cubic curves

In the diagram above, notice that there is a sharp corner where the first and second arcs are joined and there is a sharp corner where the three and fourth arcs are joined, but there is no apparent corner where the second and third arcs are joined. Even though the individual Bezier curves are smooth, piecewise Bezier curves do not have to be smooth. At the joint $p_n^1 = p_0^2$, the piecewise Bezier curve can have a sharp corner as illustrated in the diagram above. To define a piecewise Bezier curve, so that it does not have a sharp corner, we consider the value of the derivative for each curve at the joint $p_i(1) = pi + 1(0)$ (using the uniform knot sequence see below), the values of the derivatives $p_i'(1)$ and $p_{i+1}'(0)$. Recall from the previous notes that the derivative of a Bezier curve

$$b(t) = \sum_{i=0}^{n} B_i^n(t)\, p_i$$

is the Bezier curve

$$b'(t) = \sum_{i=0}^{n-1} n\, B_i^{n-1}(t)\, \Delta_i p,$$

where $\Delta_i p = p_{i+1} - p_i$ are the first differences of the control points. Therefore using a uniform knot sequence that is $t_0 = 0$, $t_1 = 1$, $t_2 = 2$ et cetera, we have that

$$b_1'(1) = n\,(p_n^1 - p_{n-1}^1) = n\,(p_1^2 - p_0^2) = p_2'(0)$$

from $B_0^n(0) = 1$ and $B_i^n(0) = 0$ if $i > 0$, and $B_i^n(1) = 0$ if $i < n$ and $B_n^n(1) = 1$. We thus conclude from $p_0^2 = p_n^1$ that $p_0^2 = p_n^1 = \frac{1}{2}(p_{n-1}^1 + p_1^2)$. Therefore, to construct a piecewise Bezier curve using two curves we give n points for each curve, and define the joint point as the midpoint of the the segment p_{n-1}^1 and p_1^2. An example of such a joining of two Bezier curves to produce a piecewise defined Bezier curve is given below.

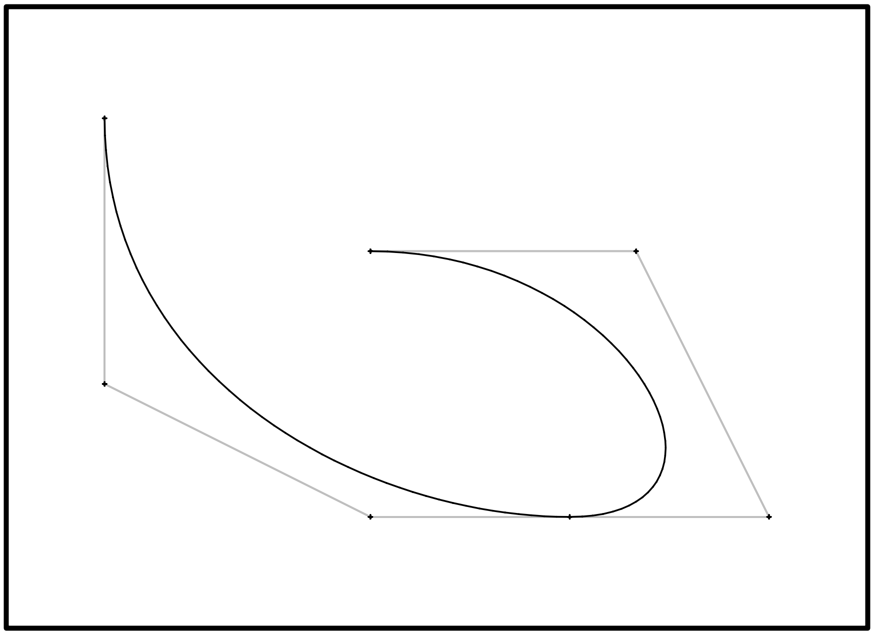

Figure 2: Construction of cubic C^1 Bezier spline of two segments

The procedure above generalizes to construct C^1 Bezier splines. For the remainder of this chapter we will mainly consider piecewise cubic Bezier curves, which we will consider in detail in the next section. But first, we note that there is no reason that we have to use $0 \leq t \leq 1$ as the interval over which the first Bezier curve is defined and there is no reason that we have to use $1 \leq t \leq 2$ as the interval over which the second Bezier curve is defined. We can instead define the composite curve over the arbitrary subintervals. This is an important observation that will be exploited when defining B-splines and is important in animation, at it allows one to use time explicitly in controlling the motion and designing the path of the motion.

To define a piecewise Bezier curve, we thus technically require in addition to control points *a knot sequence* $t_0 < t_1 < t_2 < \cdots < t_L$ where L is the number of arcs used to create the piecewise Bezier curve. The knot sequence $t_0 = 0$, $t_1 = 1$, $\cdots$, $t_L = L$ is called the *uniform knot sequence*, and if no knot sequence is given you are to assume that the uniform knot sequence is used in creating a piecewise Bezier curve. To create a Bezier curve using a nonuniform knot sequence, we first consider the linear transformation that maps the interval $t_0 \leq t \leq t_1$ to $0 \leq \tau \leq 1$, that is $\tau = (t - t_0)/(t_1 - t_0)$. We then use $(t - t_0)/(t_1 - t_0)$ as the parameter in the Bernstein polynomials defining the Bezier curve, to define the curve $c_1(t)$ can be defined as

$$c_1(t) = \sum_{i=0}^{n} B_i^n \left(\frac{t - t_0}{t_1 - t_0} \right) p_i^1 = b_1\left(\frac{t - t_0}{t_1 - t_0}\right),$$

where b_1 is the Bezier curve with control points $\{p_i^1\}$. The second arc of the piecewise Bezier curve is likewise defined as

$$c_2(t) = \sum_{i=0}^{n} B_i^n \left(\frac{t - t_1}{t_2 - t_1} \right) p_i^2 = b_2(\frac{t - t_1}{t_2 - t_1}).$$

This can be extended to other other arcs defined on $t_i \leq t \leq t_{i+1}$ in the obvious manner.

We note the parameterization of the curve has no affect on the zeroth order continuity of a piecewise Bezier curve. The equality of the first and last control points determine the zeroth order continuity. In fact, the fact that $c_i(t) = b_i((t - t_{i-1})/(t_i - t_{i-1}))$ implies that the parameterization of a piecewise Bezier curve has no affect on the shape of the curve, since $\tau(t) = (t - t_{i-1})/(t_i - t_{i-1})$ is a one-to-one and onto function from $[t_{i-1}, t_i]$ to $[0, 1]$, and there is a inverse $t = t_{i-1} + \tau(t_i - t_{i-1})$. Reparameterizations of a curve in general have no affect on the shape of curve, but can affect the smoothness of the curve as you are to demonstrate in one of the exercises.

Exercises

1. Play with the applet: Creating C^0 Bezier Splines. This applet uses the uniform knot sequence, as the knot sequence has no affect on the curve.

2. Where do you place the control points to approximate a circle with a C^0 Bezier spline? Use the Applet: Fitting a C^0 Bezier spline to a circle to investigate this. How does the number of segments change the configuration of the control points and the control polyline?

3. Given the control polyline below sketch the C^0 cubic Bezier spline.

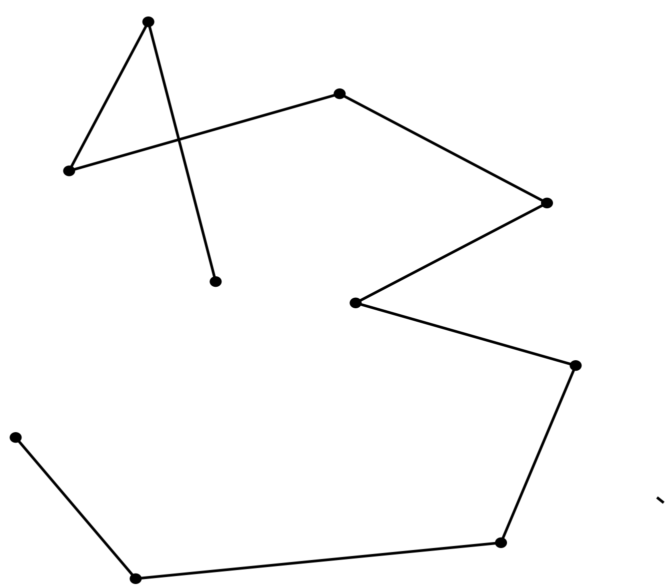

Figure 3: Construct C^0 cubic Bezier spline

4. Given the control points below, determine if the C^0 cubic Bezier spline will be differentiable with the uniform knot sequence.

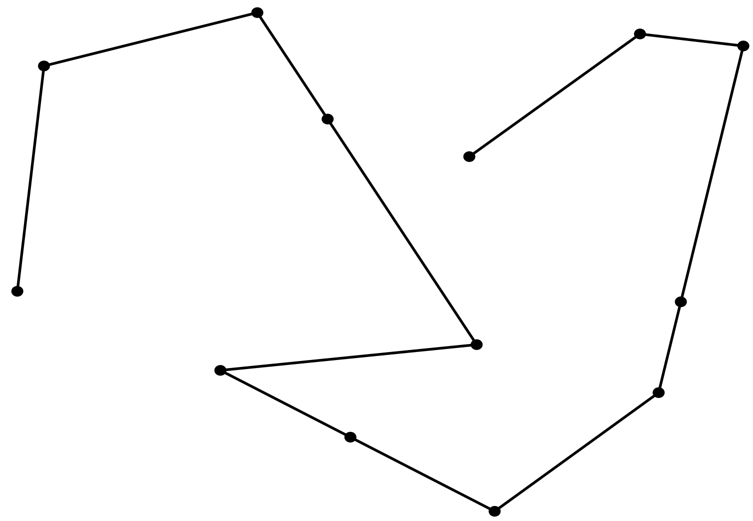

Figure 4: Is this C^0 Bezier spline differentiable?

5. Determine the conditions required for a piecewise Bezier curve with two cubic curves with control points p_0^k, p_1^k, p_2^k and p_3^k ($k = 1$ or $k = 2$) using the nonuniform knot sequence t_0, t_1, t_2 to be C^1, that is have a continuous derivative at the joint point $p_3^1 = p_0^2$. This means compute $b_1'(t_1)$ and $b_2'(t_1)$ where $b_1(t) = \sum B_i^3((t-t_0)/(t_1-t_0))\, p_i^1$ and $b_2(t) = \sum B_i^3((t-t_1)/(t_2-t_1))\, p_i^2$, and determine the relations between p_2^1, $p_3^1 = p_0^2$, p_1^2.

17.2 C^1 cubic Bezier Splines

We now start considering curves that truly are Bezier splines. Our principal interest in this section is with C^1 cubic Bezier splines. Cubic Bezier splines are of special interest as they allow us to define and approximate curves very well with simple curves. The approximation of a circle by using cubic curves to approximate a quarter circle in the interactive exercises is a simple example of the use of cubic Bezier splines.

Let us start by considering, a piecewise Bezier curve consisting of three cubic Bezier curves. To define this piecewise curve, we need 10 control points. Each cubic curve requires 4 points for a total of 12 points, but there are two joint points, meaning we need 10 control points. If we desire the curve to smooth at the joint points, we really only need 8 control points, as the joint points are defined by the two nearest points. The first three control points of the first cubic curve segment, the last three control points of the third cubic curve segment, and the middle two control points of the second curve segment, see the diagram below.

In general, we need to be given $2L+2$ points to define a C^1 cubic Bezier spline (with the uniform knot sequence) that is made up of L cubic Bezier arcs. This is an advantage over the number of points that must be given to create a C^0 cubic Bezier spline, as $3L+1$ points need to be given to create a C^0 cubic Bezier spline. The exact procedure for producing the Bezier spline is given below. We start with the spline control points d_{-1}, d_0, d_1, $\cdots$, d_{2L}. [The index in defining the spline control points begins at -1 so that the last spline control

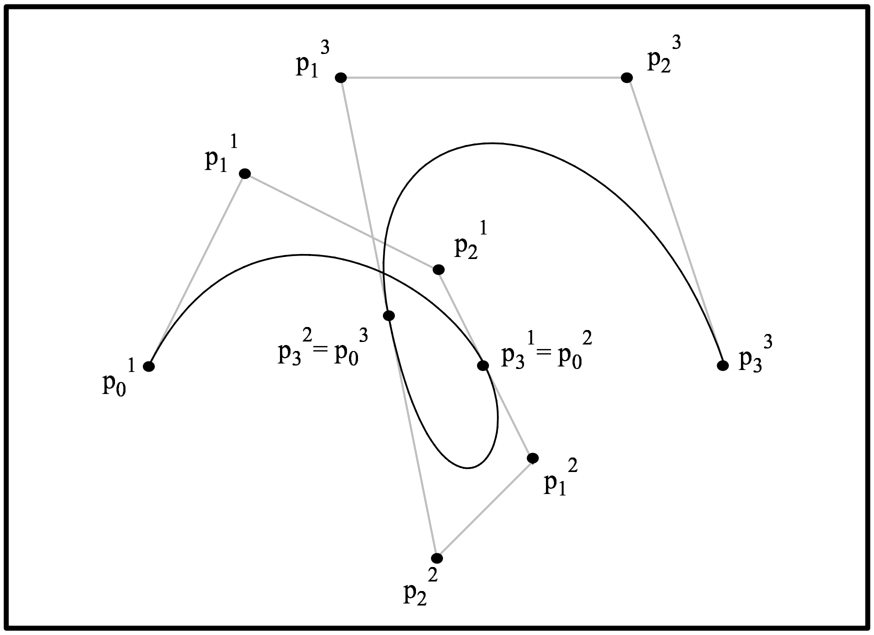

Figure 5: A differentiable piecewise Bezier curve of three segments

point is given by $2L$ from which it is easy to determine the number of curve segments.] To produce, the L cubic Bezier curves we need $4L$ control points, however since the spline is to be continuous $L-1$ of these points are joints (the points where consecutive cubic curves meet), that means we really need to define $3L+1$ control points from the $2L+2$ spline control points. From the $3L+1$ control points p_0, p_1, p_2, $\cdots$, p_{3L}, we define the individual cubic Bezier curves $b_j(t)$ with $j = 1, 2, \cdots, L$ as

$$b_j(t) = (1-t)^3\, p_{3j-3} + 3t(1-t)^2\, p_{3j-2} + 3t^2(1-t)\, p_{3j-1} + t^3\, p_j$$

The procedure for defining the control points p_i from the given control points d_{-1}, d_0, d_1, $\cdots$, d_{2L} and a uniform knot sequence is to set

$$p_0 = d_{-1}$$
$$p_1 = d_0$$
$$p_2 = d_1$$

to start the definition of the control points of the Bezier curves. Then we define,

$$p_{3i+1} = d_{2i} \quad \text{and} \quad p_{3i+2} = d_{2i+1} \quad \text{for} \quad i = 1, 2, \cdots, L-1.$$

This gives the middle two points of each intermediate Bezier curve, and all but the final point in the last Bezier curve. We then define $p_{3L} = d_{2L}$ (the last control point in defining the piecewise Bezier curve = the last control point in the Bezier spline), and then we define

$$p_{3i} = \frac{1}{2}\,(p_{3i-1} + p_{3i+1}) \quad \text{with} \quad i = 1, 2, \cdots, L-1,$$

which gives the joint points of the Bezier curves. See diagram, below.

To use a nonuniform knot sequence, the points p_{3i} where two cubic Bezier curves meet in the spline no longer need to be at the midpoint of the segment p_{3i-1} and p_{3i+1} that are defined

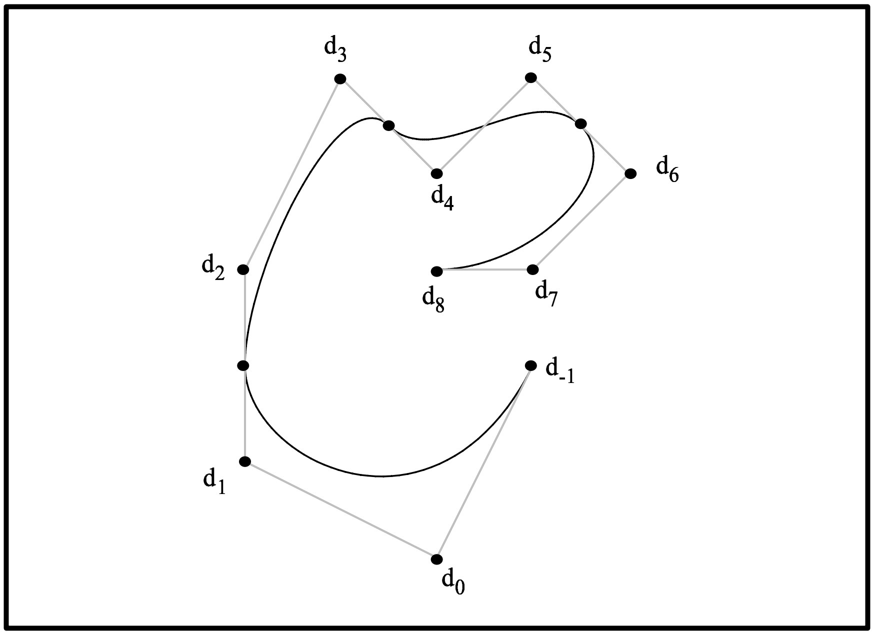

Figure 6: A C^1 Bezier spline of 4 segments with its spline control points.

directly from the control points d_k. It does have to be on the line segment $p_{3i-1}p_{3i+1}$ but not necessarily at the midpoint. The exact location is determined by the knot sequence, see exercises below.

Exercises

1. Play with the applet: Creating C^1 (cubic) Bezier Splines. Notice the difference between a C^1 Bezier spline and the Bezier curve with the same control points. In particular, play with how the control points affect the shape of the curve. This applet uses uniform knot sequences.

2. Where do you place the control points to approximate a circle with a C^1 Bezier spline? Use the Applet: Fitting a C^1 Bezier spline to a circle to investigate this. How does the number of segments change the configuration of the control points and the control polyline?

3. Determine the algorithm to create an nth degree C^1 Bezier spline, a piecewise Bezier curve consisting of nth order Bezier curves.

4. Extend the conditions from Exercise 4 in the previous subsection to define a cubic Bezier spline with L cubic arcs and a nonuniform knot sequence.

5. Determine the conditions relating the control points of an nth degree C^1 Bezier spline to the associated piecewise Bezier curve using a non-uniform knot sequence.

6. Sketch the C^1 Bezier spline given the control polylines below (assuming a uniform) sequence

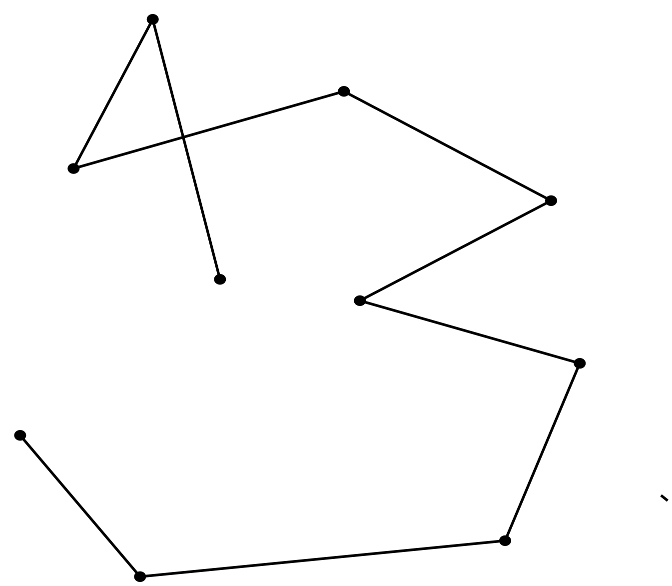

Figure 7: Sketch the C^1 Bezier spline given the control polyline

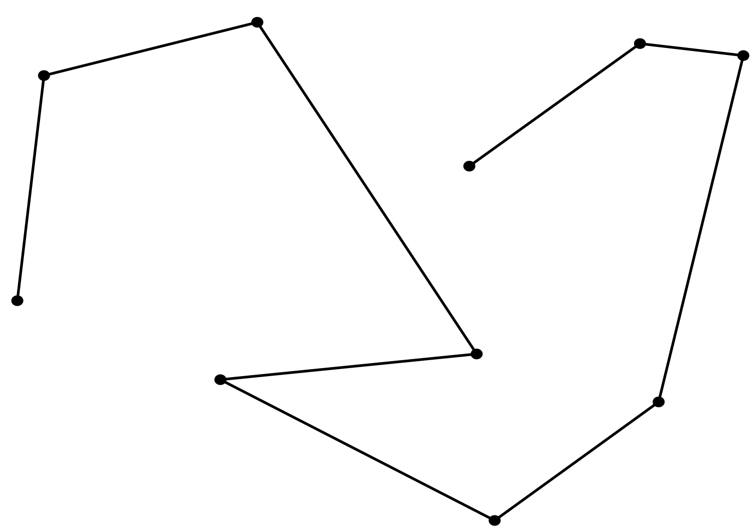

Figure 8: Sketch the C^1 Bezier spline given the control polyline