B-spline Lab

An interactive MATLAB tool for B-spline curves

2015, P.J. Barendrecht

1 Introduction

[Todo]

$$C(t) = \sum_i P_i B_i(t)$$

Controls (mouse and keyboard) are highlighted in grey, e.g. `right click` or `A`.

2 Example

Run `BsplineLab.m` without providing any arguments. By default, a cubic B-spline curve[1] will be drawn once 5 control points[2] have been positioned (`left click`).

2.1 Basics

Each control point is assigned a label P_i. Next, the control points are connected by line-pieces (edges). The control points and edges constitute what is called the *control polygon*, often referred to as the *control net*. See Figure 2.1.

Before modifying the curve and looking at the other options B-spline Lab has to offer, let us start by taking a closer look at the current curve. The first step is to display the basis functions (`B`) associated with the default knot-vector[3] $\Xi = [0\ 1\ 1\ 1\ 2\ 3\ 3\ 3\ 4]$. Each control point P_i is associated with a B-spline basis function $B_i(t)$ which is visualised using colours. See Figure 2.2.

Each B-spline basis function[4] is piecewise polynomial (concatenated polynomial pieces) joining with – in general – optimal parametric continuity []. The resulting parametric curve inherits these properties.

Each part of the B-spline basis spans the complete space of cubic polynomials[5]. In other words, at each appropriate parameter value 4 B-splines are defined. From Figure 2.2 it follows that the curve consists of two segments. The first segment corresponds to

[1] Cubic curves are intuitive to work with for several reasons [].

[2] This way the curve will consist of two segments, providing us with the possibility to study the continuity of the connection of the two segments.

[3] In short, a non-decreasing sequence of numbers in parameter space.

[4] *"B-spline basis function"* is a bit redundant as the *B* in *B-spline* stands for *basis*.

[5] This space can be spanned by different bases, for instance by $< 1, x, x^2, x^3 >$.

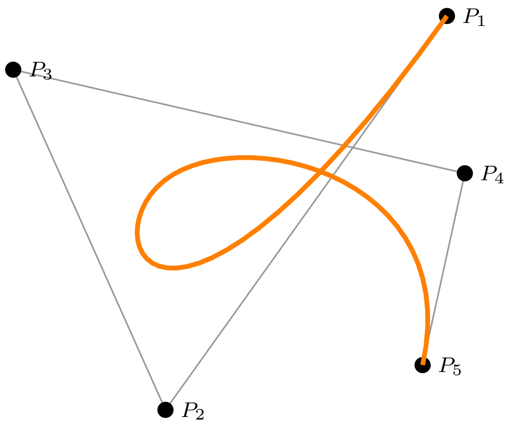

Figure 2.1: *Cubic B-spline curve (orange) and control net (black/grey). Note that the first and last control points (P_1 and P_5) are interpolated, whereas the others are only approximated.*

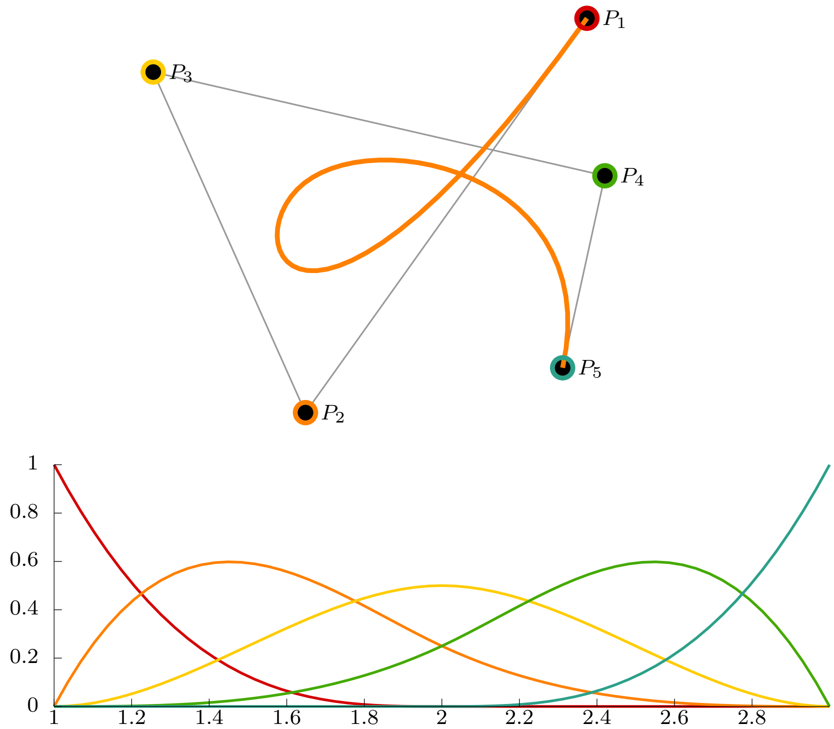

Figure 2.2: *B-spline functions (* `B` *) for $d = 3$ (cubic) and $\Xi = [0\ 1\ 1\ 1\ 2\ 3\ 3\ 3\ 4]$. Each basis function is associated with a control point. For example, the dark-red function is associated with P_1, the green one with P_4.*

$t \in [1, 2)$, the second one to $t \in [2, 3)$. It can be useful to plot the endpoints of these (half-open) intervals (`J`). These points are the images of knots and are sometimes referred to as *joints*. See Figure 2.3.

Close plot of the B-spline basis functions (`Q`).

2.2 First derivative, tangent continuity

First derivative (tangent direction and length) []

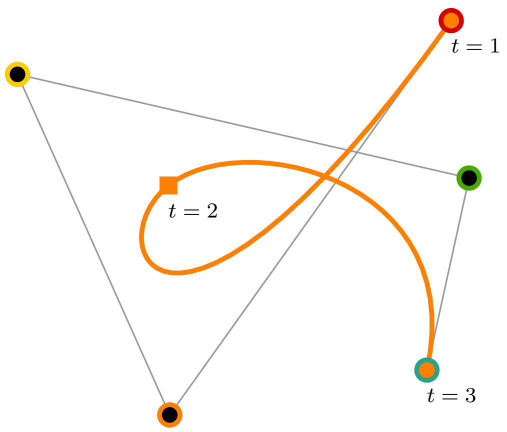

Figure 2.3: *Joints (`J`) of the cubic B-spline curve plotted as solid squares. The control point labels have been disabled (`L` several times).*

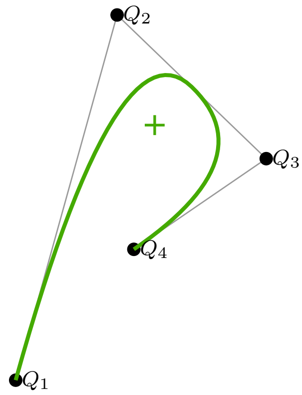

Figure 2.4: *Hodograph (`H`) of the cubic B-spline curve from Figure 2.3. The "+" indicates the origin.*

2.3 Adding control points

Information () on the current curve can be displayed (`I`). Change the knot-vector from [011123334] to [01112345556] (`K`).

2.4 Evaluating the curve at arbitrary parameter values

Evaluating...

2.5 Moving control points

2.6 Changing knot-spans (local approach)

Change selection mode to edges (`E`).

2.7 Changing weights

Change selection mode back to vertices (`V`).

Reset all weights to unity (`U`).

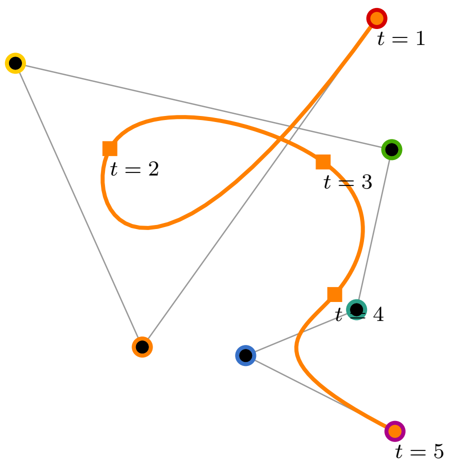

Figure 2.5: *Cubic B-spline, now with two more control points (and hence two more segments).*

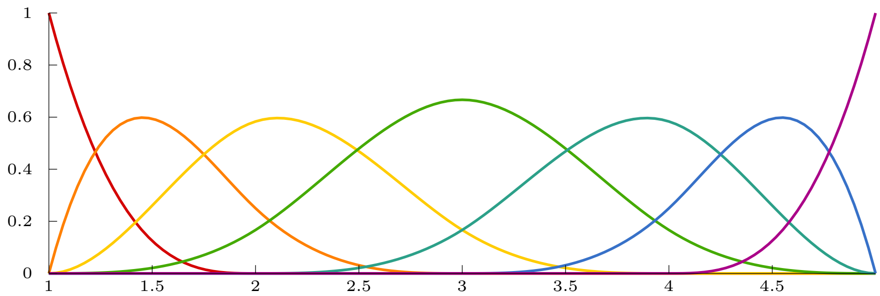

Figure 2.6: *B-splines associated with the new knot-vector. Once again, the colours correspond to the colours of the control points (in Figure 2.5).*

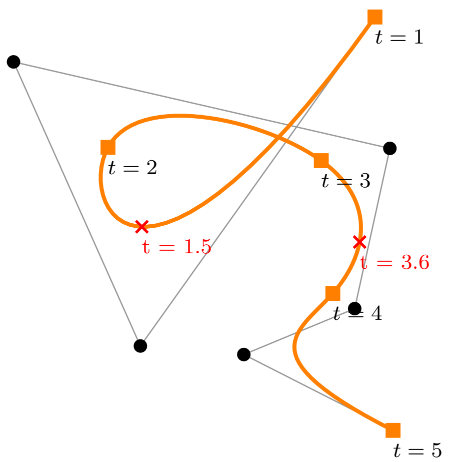

Figure 2.7: *Evaluation (* `T` *) of the B-spline curve at two arbitrary parameter values. The results are indicated with a red "x".*

2.8 Changing the knot-vector (global approach)

Define new knot-vector (`K`), $\Xi = [0112345667]$.

:: Deleted 1 control point(s).

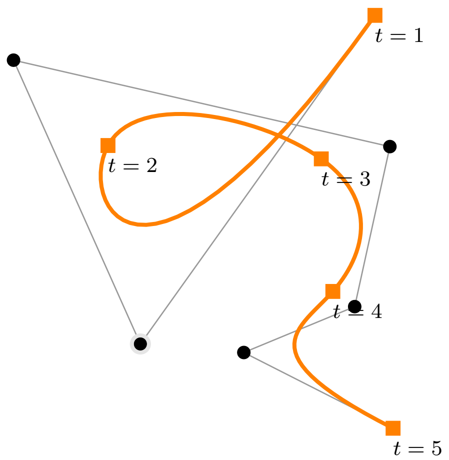

Figure 2.8: *The second control point (P_2) is selected (* `right click` *). This is indicated using a light-grey circle.*

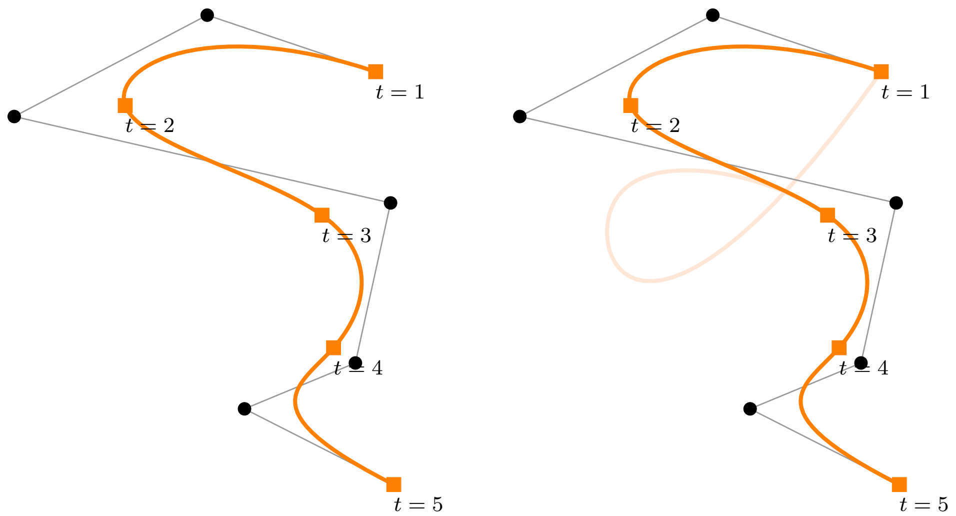

Figure 2.9: *The selected control point is re-positioned (* `G` *and* `left click` *), resulting in a modified curve (left). The affected segments of the curve correspond to $t \in [1, 3)$ as shown on the right.*

2.9 Changing the degree

`D`

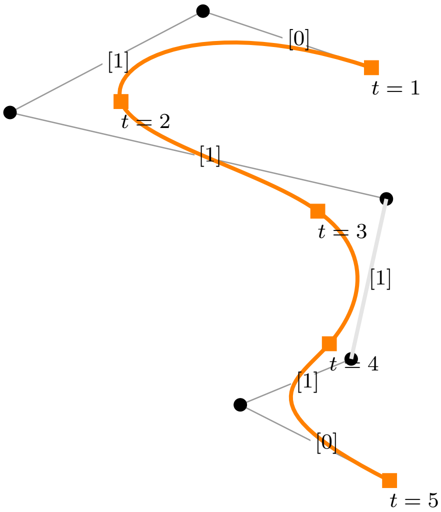

Figure 2.10: *An edge of the control net is selected (* `right click` *). The edge corresponds to a segment with knot-span 1 as indicated in square brackets.*

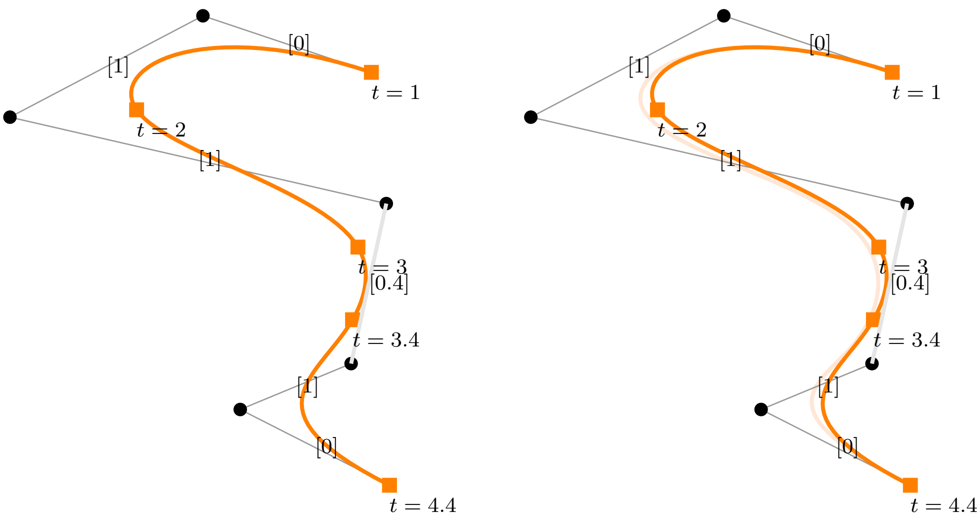

Figure 2.11: *The knot-span is decreased (* `-` *), resulting in a modified curve (left). The previous and current curves are compared (right).*

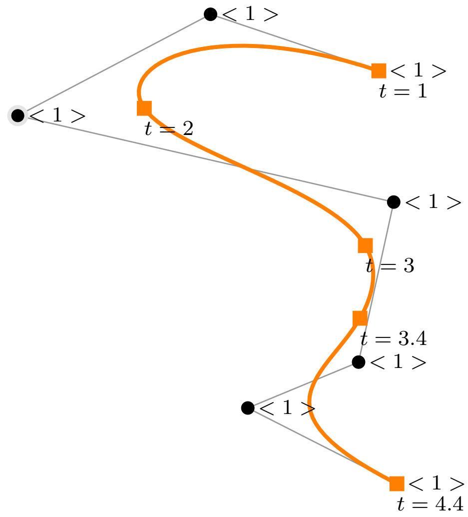

Figure 2.12: *The third control point (P_3) is selected (* `right click` *). The weights are displayed in angle brackets.*

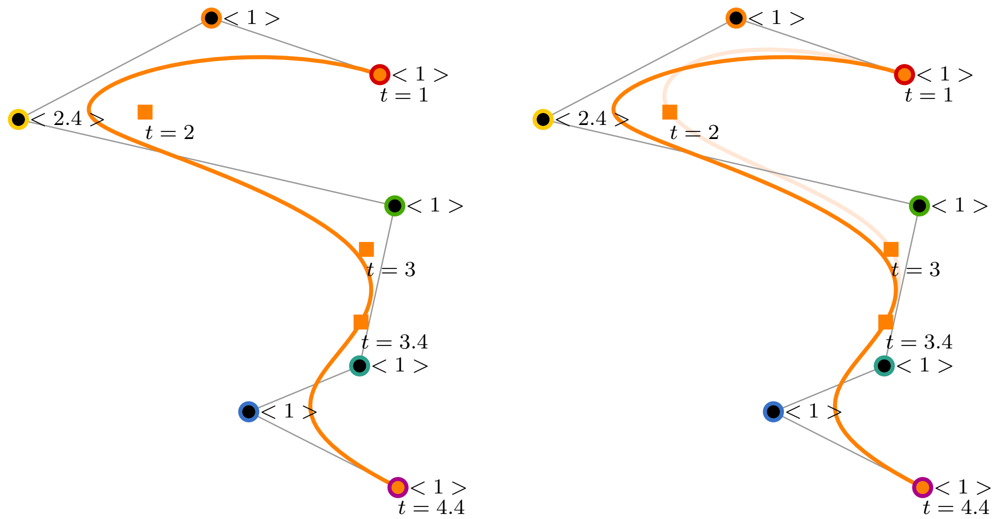

Figure 2.13: *The weight is increased (* `scroll up` *), resulting in a modified curve (left). The old and new curves are compared (left).*

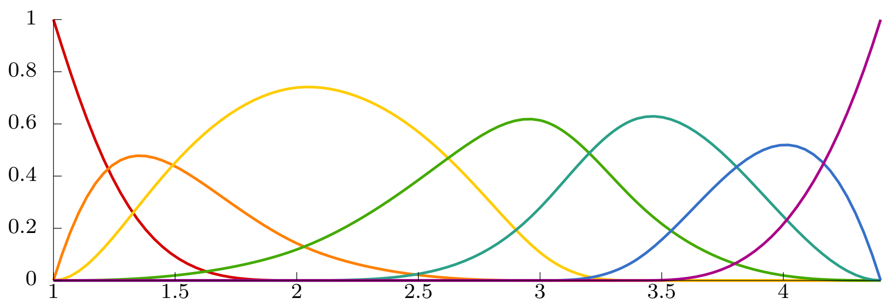

Figure 2.14: *The cubic B-spline basis corresponding to the current knot-vector and set of weights. As neither the knot-spans nor the weights are uniform anymore we now have a NURBS basis.*

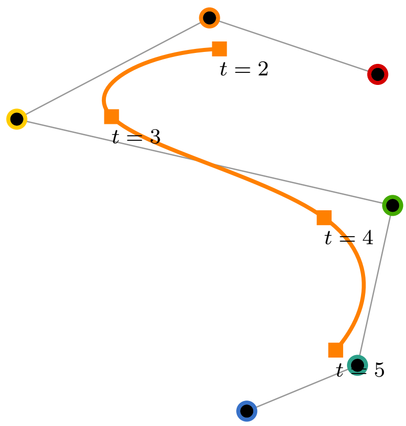

Figure 2.15: *Cubic B-spline curve after modifying the global knot-vector directly (* `K` *). The end-points are no longer interpolated.*

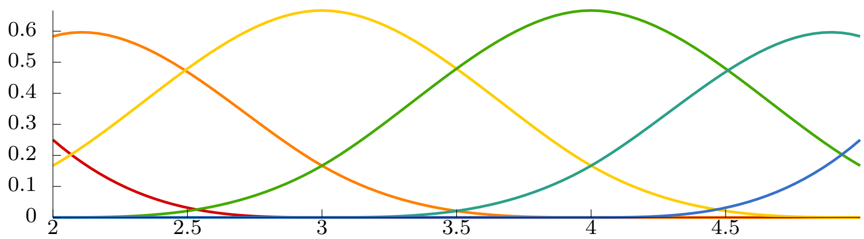

Figure 2.16: *Basis associated with the previous curve.*

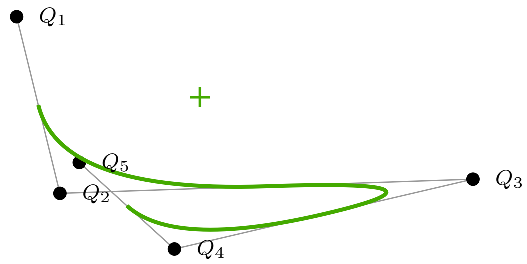

Figure 2.17: *Hodograph (* `H` *).*

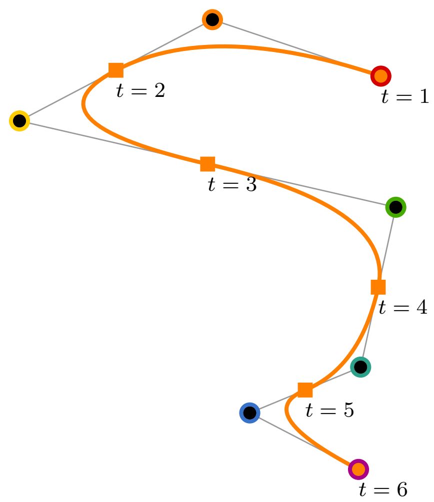

Figure 2.18: *Quadratic B-spine curve.*

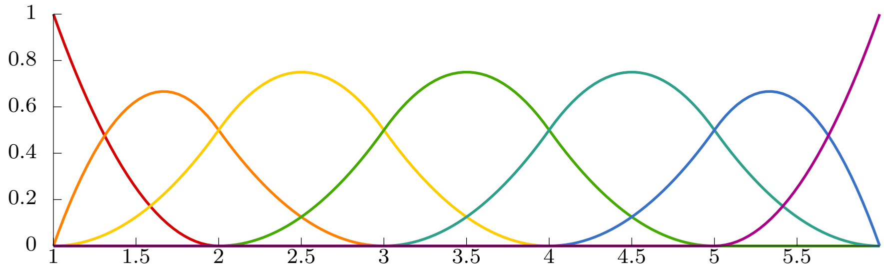

Figure 2.19: *Quadratic B-splines.*

```
|-----||-----|
      ||-----|-----|
       |-----|-----|-----|
             |-----|-----|-----|
                   |-----|-----|-----|
                         |-----|-----||
                               |-----||-----|
```

Figure 2.20: *Schematic overview (* `F` *) of the supports of the basis funcctions and their overlap. It follows that the quadratic curve consists of 5 segments.*

3 Extra

3.1 Basis

Plot basis on entire parameter domain (`Shift + B`)

3.2 Polar forms

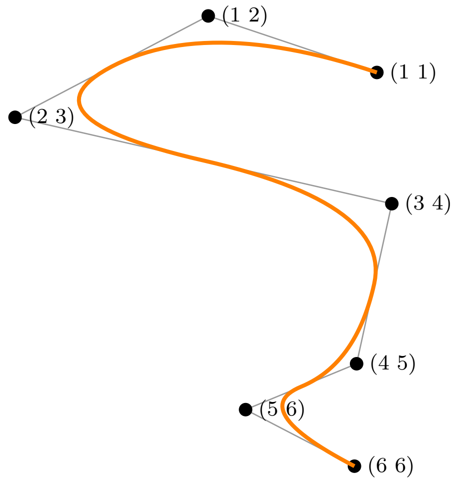

Figure 3.1: *Blossom labels (* `L` *several times).*

3.3 Drawing part of a circle

Toggle axis visibility (`A`)
Snap to grid (`#`) when re-positioning control points
Restrict movement of control points to X-axis (`X`) or Y-axis (`Y`)
Set exact position of selected control point (`P`)
Set exact weight of selected control point (`W`)

3.4 Misc

Saving[6] the curve and/or basis as scalable vector graphic (`S`)
Displaying help (`?`) [Todo]

[6] Requires external package []