The Bernstein-Bézier Form of a Polynomial[−1−]

Note: This document is meant to be viewed in color. If you only have a black and white version you can download a colored version at `http://www.math.utah.edu/~pa/bform.pdf`

Introduction.

There are many ways to write a polynomial. One particular representation is the **Bernstein-Bézier** form (or *B-form* for short) of a polynomial whose use has become extremely wide spread during the past two decades or so. The reason for this well grounded popularity is the fact that the B-form allows to approach algebraic problems (such as evaluation, but in particular, the smooth joining of polynomial pieces) geometrically.

One variable.

We begin by representing a univariate polynomial in a form that will at first appear cumbersome, but that will lead to a natural generalization in the case of two or more variables.

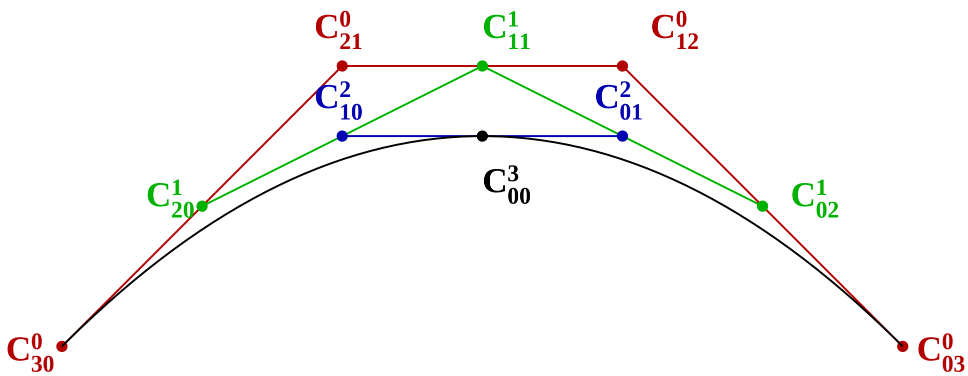

Figure 1. A cubic Bézier Curve.

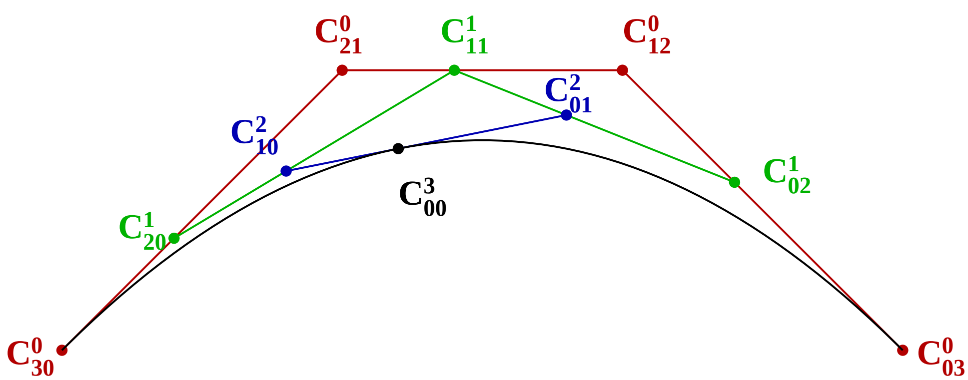

Figure 2. Evaluation at another point.

−1− TEX processed June 9, 2014 by Peter Alfeld, Department of Mathematics, University of Utah, Salt Lake City, Utah 84112, U.S.A., JWB 127, 801-581-6842, pa@math.utah.edu, `http://www.math.utah.edu/~pa/`

Let p be a polynomial of degree d defined on an interval $I = [V_1, V_2]$ where $V_2 \neq V_1$. We express a point P in I as

$$P = b_1 V_1 + b_2 V_2 \qquad \text{where} \qquad b_1 + b_2 = 1. \tag{1}$$

b_1 and b_2 are the *barycentric coordinates* of P with respect to the interval I. They are uniquely defined by the equations (1). In fact,

$$b_1 = \frac{V_2 - P}{V_2 - V_1} \quad \text{and} \quad b_2 = \frac{P - V_1}{V_2 - V_1} \tag{2}$$

We write p in its *Bernstein-Bézier Form* (or just *B-form*) as

$$p(P) = \sum_{i+j=d} \frac{d!}{i!j!} c_{ij} b_1^i b_2^j. \tag{3}$$

It is always tacitly understood in B-form lore that the subscripts are non-negative integers. The coefficients c_{ij} are the *Bézier-ordinates* of p (or, more casually, just the *coefficients*). Associated with the coefficients are their *Domain Points*

$$P_{ij} = \frac{iV_1 + jV_2}{d} \tag{4}$$

The domain points are combined with the Bézier ordinates to form the *Control Points* $C_{ij} = (P_{ij}, c_{ij}) \in R^2$ which together form the *Control Polygon.* Figures 1 and 2 shows the control polygon and the control points of a cubic polynomial in red. (The other ingredients of the Figure are explained below.)

Example. *Consider the special case*

$$[V_1, V_2] = [0, 1], \quad P = x, \quad b_1 = (1 - x), \quad b_2 = x, \quad \textit{and} \quad c_{ij} = f\left(\frac{j}{n}\right). \tag{5}$$

We get

$$p(x) = B_n(f, x) = \sum_{k=0}^{n} f\left(\frac{k}{n}\right)\binom{n}{k} x^k (1 - x)^{n-k} \tag{6}$$

which is the Bernstein polynomial in Bernstein's proof of the Weierstrass Approximation Theorem. This explains why Bernstein's name occurs in the phrase Bernstein-Bézier Form. Sergei Bernstein (1880-1968) was a mathematician. Pierre Bézier(1910-1999), working as an engineer for the French firm Renault, developed the Bernstein-Bézier Form as a tool for the design of surfaces (such as the shape of a vehicle).

The univariate de Casteljau Algorithm

The polynomial p can be evaluated at $P = b_1 V_1 + b_2 V_2$ by the **de Casteljau Algorithm**. Paul de Casteljau (born 1930) developed this algorithm in 1959 while working as an engineer for the French car company Citroën.

$$\begin{aligned} p(P) \quad &= \quad c_{00}^d \quad \text{where} \\ c_{ij}^0 \quad &= \quad c_{ij} \qquad\qquad\qquad\qquad i + j = d \\ \text{and} \quad c_{ij}^{k+1} \quad &= \quad b_1 c_{i+1,j}^k + b_2 c_{i,j+1}^k, \quad i + j + k + 1 = d, \qquad k = 0, \ldots, d - 1. \end{aligned} \tag{7}$$

Note that the superscripts of c indicate the stage of the algorithm, they are not exponents.

To see that this works we show by induction that

$$p(P) = \sum_{i+j=d-k} \frac{(d-k)!}{i!j!} c_{ij}^k b_1^i b_2^j \tag{8}$$

for all $k = 0, 1, \ldots d$. Clearly this is true for $k = 0$. Suppose it is true for $k < d$. Then we obtain

$$\begin{aligned}
\sum_{i+j=d-k-1} \frac{(d-k-1)!}{i!j!} c_{ij}^{k+1} b_1^i b_2^j &= \sum_{i+j=d-k-1} \frac{(d-k-1)!}{i!j!} (b_1 c_{i+1,j}^k + b_2 c_{i,j+1}^k) b_1^i b_2^j \\
&= \sum_{\mu+\nu=d-k} \frac{(d-k-1)!}{\mu!\nu!} (\mu c_{\mu\nu}^k + \nu c_{\mu\nu}^k) b_1^\mu b_2^\nu \\
&= \sum_{\mu+\nu=d-k} \frac{(d-k-1)!(d-k)}{\mu!\nu!} c_{\mu\nu}^k b_1^\mu b_2^\nu \\
&= \sum_{i+j=d-k} \left[\frac{(d-k)!}{i!j!} c_{ij}^k \right] b_1^i b_2^j
\end{aligned} \tag{9}$$

The de Casteljau Algorithm can be run with the control points instead of the coefficients. Setting

$$C_{ij}^0 = C_{ij} \quad \text{and} \quad C_{ij}^{k+1} = b_1 C_{i+1,j}^k + b_2 C_{i,j+1}^k$$

we get

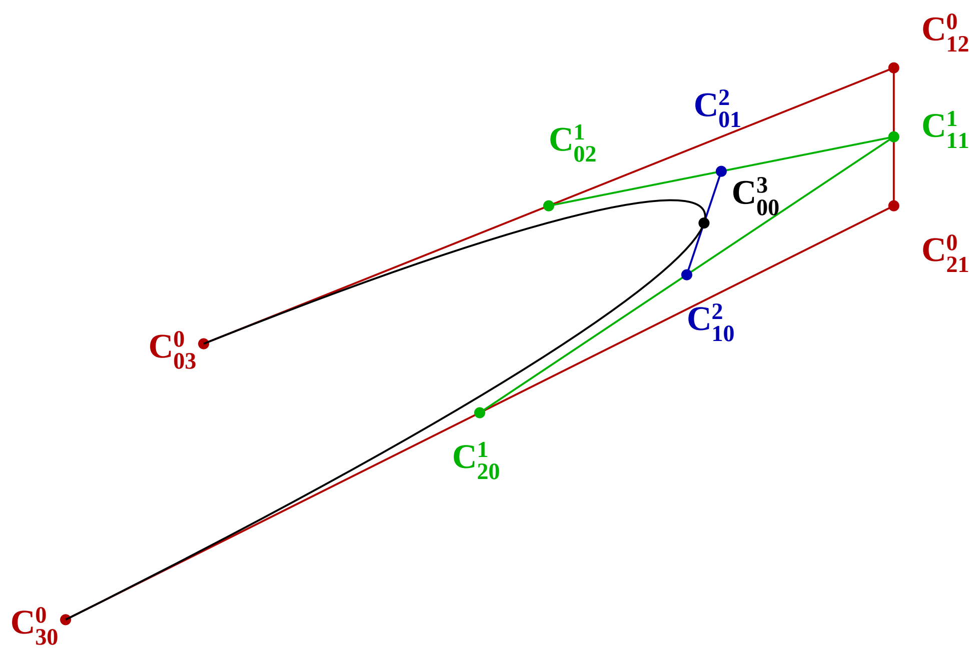

Figure 3. A parametric cubic Bézier Curve.

$$\begin{array}{llll}
p(P) = C_{00}^d \quad \text{where} & & & \textbf{de Casteljau Algorithm} \\
& C_{ij}^0 = C_{ij} & i+j=d & \\
\text{and} & C_{ij}^{k+1} = b_1 C_{i+1,j}^k + b_2 C_{i,j+1}^k, & i+j+k+1=d, & k=0,\ldots,d-1.
\end{array} \tag{10}$$

This version is shown (for the case $d = 3$) in Figures 1 (where $b_1 = b_2 = 0.5$) and 2 (where $b_1 = 0.6$ and $b_2 = 0.4$). Note that the control polygons (drawn in red) are identical for the two Figures. Just the point of evaluation, and hence the stages of the de Casteljau algorithm are different.

The version (10) can be applied to the *parametric case* where the C_{ij}^0 are any points in $\mathbb{R}^2$, as illustrated in Figure 3. In this context the resulting curve is known as a *Bézier curve*. However, in these notes we focus on the functional case.

One particularly useful feature of the B-form is that individual curves can be matched smoothly by meeting conditions that are *geometric* rather than *algebraic*.

It is clear that $p(V_1) = c_{d0}$ and $p(V_2) = c_{0d}$. Thus to match two curves continuously we only have to assure that the control polygons touch at the end points. What about matching derivatives?

Using a prime to denote a derivative with respect to P we obtain from (1) that

$$b_2' = -b_1' = \frac{1}{V_2 - V_1} \tag{11}$$

Differentiating in (3) gives

$$\begin{aligned} p'(P) &= \sum_{i+j=d} \frac{d!}{i!j!} c_{ij} \left(i b_1' b_1^{i-1} b_2^j + j b_1^i b_2' b_2^{j-1} \right) \\ &= \sum_{i+j=d-1} \frac{1}{V_2 - V_1} \left[-\frac{(i+1)d!}{(i+1)!j!} c_{i+1,j} + \frac{(j+1)d!}{i!(j+1)!} c_{i,j+1} \right] b_1^i b_2^j \\ &= \sum_{i+j=d-1} \frac{1}{V_2 - V_1} \left[-\frac{d!}{i!j!} c_{i+1,j} + \frac{d!}{i!j!} c_{i,j+1} \right] b_1^i b_2^j \\ &= \sum_{i+j=d-1} \frac{(d-1)!}{i!j!} \hat{c}_{ij} b_1^i b_2^j \end{aligned} \tag{12}$$

where

$$\hat{c}_{ij} = \frac{d(c_{i,j+1} - c_{i+1,j})}{V_2 - V_1}. \tag{13}$$

Thus we have written the derivative of p as a polynomial of degree $d-1$ in Bernstein-Bézier Form. Evaluating at the end points gives:

$$p'(V_1) = \frac{d(c_{d-1,1} - c_{d,0})}{V_2 - V_1} \quad \text{and} \quad p'(V_2) = \frac{d(c_{0,d} - c_{1,d-1})}{V_2 - V_1}. \tag{14}$$

Thus the slope of the Bézier curve at the end points equals the slope of the segment at the end of the control polygon. This means that two Bézier curves will join differentiably if the two touching segments of the two control polygons are collinear!

This is illustrated in Figure 4 for the functional case and in Figure 5 for the parametric case.

Higher Order Smoothness Conditions

As an exercise you may want to explore higher order smoothness conditions. It turns out that two curves join smoothly of order $r \geq 1$ if the degree r polynomial defined by the points C_{d0}^0, $C_{d-1,0}^1$, … $C_{d-r,0}^r$ is the same polynomial as the polynomial defined by the corresponding points of the neighboring control polygon.

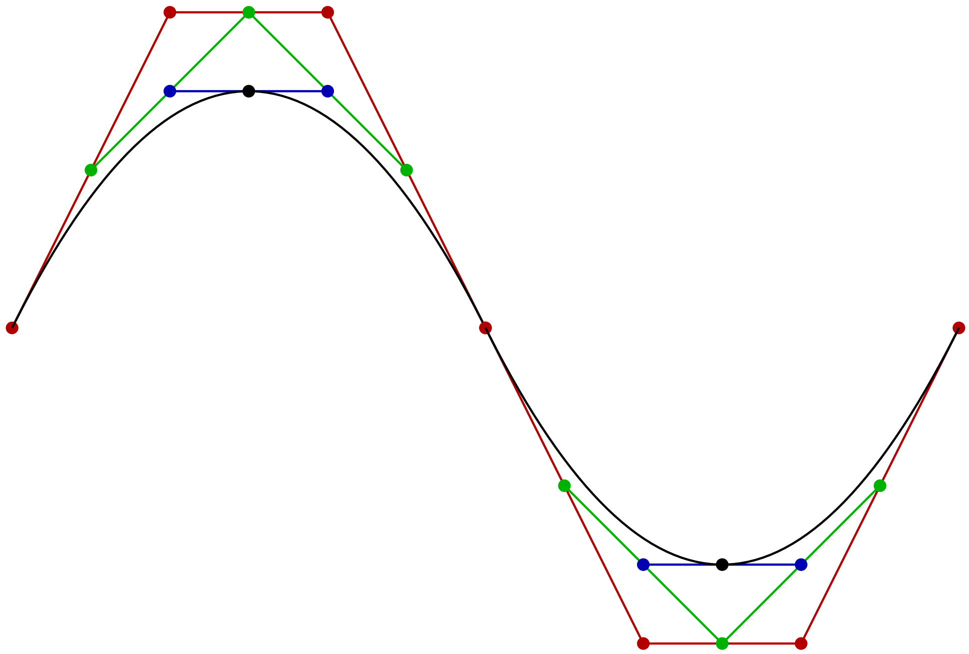

Figure 4. Joining two functional Bézier segments.

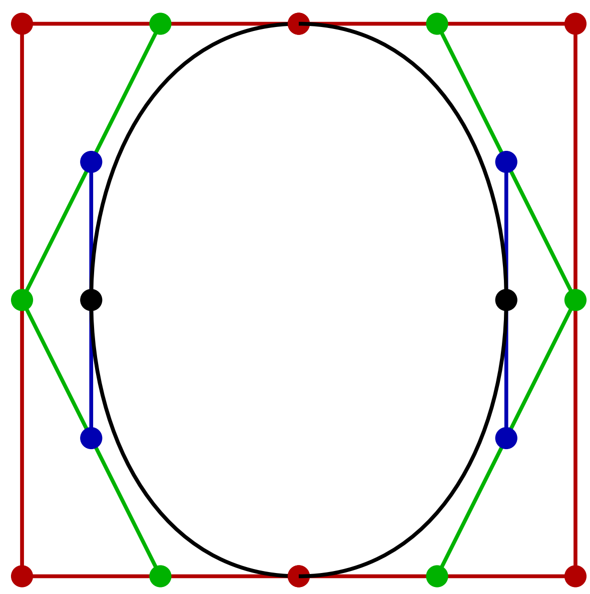

Figure 5. Joining two parametric Bézier curves.

Subdivision, Rendering

Consider the two polynomials defined by the control points C_{d0}^0, $C_{d-1,0}^1$, $\ldots$ $C_{0,0}^d$ and $C_{0,0}^d$, $C_{0,1}^{d-1}$, $\ldots$ C_{0d}^d, respectively. It turns out (exercise) that these two polynomials are identical, and equal the original polynomial p. As a consequence of this fact a Bézier polynomial can be *rendered* by repeated evaluation (or subdivision) until the resulting set of control polygon is sufficiently refined.

The two variable case

The two variable case parallels the one variable case. A bivariate polynomial of degree d in two variables is usually written as

$$p(P) = \sum_{i+j \le d} \alpha_{ij} x^i y^j. \tag{15}$$

where x and y are the cartesian coordinates of a point $P \in \mathbb{R}^2$. Such a polynomial has $\binom{d+2}{2}$ coefficients.

In the two variable case, instead of intervals we have triangles. Instead of a partition of an interval into subintervals we have a triangulation:

A set $\triangle = T_1, T_2, \ldots, T_N$ of N triangles T_i is a triangulation provided:

1. The union $\Omega = \bigcup_{i=1}^N T_i$ is a polygon.
2. The intersection of any two triangles in $\triangle$ is empty, a single vertex, or a single edge.

(In this context, the word "polygon" is meant in the ordinary sense. Its edges form a circuit and do not cross, and the polygon does not have holes.)

Given a triangulation with N triangles, let's denote by

$$\begin{aligned} V_I &= \text{number of interior vertices} \\ V_B &= \text{number of boundary vertices} \\ V &= V_B + V_I = \text{number of vertices} \\ E_I &= \text{number of interior edges} \\ E_B &= \text{number of boundary edges} \\ E &= E_B + E_I = \text{number of edges} \end{aligned} \tag{16}$$

One can show easily by induction (exercise) that

$$\begin{aligned} V_B &= E_B \\ E_I &= V_B + 3V_I - 3 \\ N &= V_B + 2V_I - 2 \end{aligned} \tag{17}$$

Let T be a triangle with the vertices V_1, V_2, and V_3. We express a point $P \in \mathbb{R}^2$ as

$$P = \sum_{i=1}^3 b_i V_i \qquad \text{where} \qquad \sum_{i=1}^3 b_i = 1. \tag{18}$$

The b_i are the *barycentric coordinates* of P with respect to T. The polynomial p is given in *Bernstein-Bézier Form* as

$$p(P) = \sum_{i+j+k=d} \frac{d!}{i!j!k!} c_{ijk} b_1^i b_2^j b_3^k. \tag{19}$$

As before, the coefficients c_{ijk} are the *Bézier ordinates* (or *coefficients*) of p. They are associated with *domain points*

$$P_{ijk} = \frac{iV_1 + jV_2 + kV_3}{d} \tag{20}$$

and *control points*

$$C_{ijk} = (P_{ijk}, c_{ijk}) \in \mathbb{R}^3. \tag{21}$$

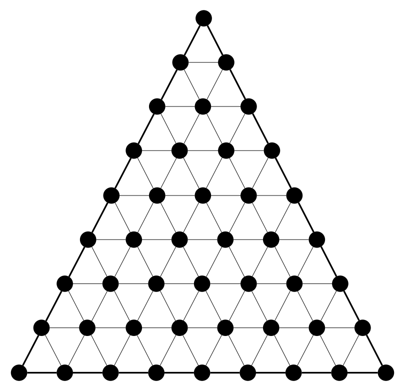

Figure 6. Domain points for $d = 8$.

Figure 6 shows the domain points on a single triangle for the case $d = 8$.

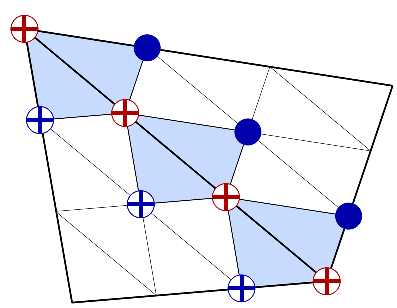

Figure 7. Smoothness Conditions.

The bivariate de Casteljau Algorithm

A bivariate polynomial can be evaluated by a straightforward modification of the univariate algorithm (7)

$$\begin{aligned} p(P) &= c_{000}^d \quad \text{where} \\ c_{ijk}^0 &= c_{ijk} \qquad\qquad i+j+k=d \\ \text{and} \quad c_{ijk}^{m+1} &= b_1 c_{i+1,j,k}^m + b_2 c_{i,j+1,k}^m + b_3 c_{i,j,k+1}^m, \quad {\scriptstyle i+j+k+m+1=d \atop m=0,\ldots,d-1} \end{aligned} \tag{22}$$

Bivariate smoothness conditions

Smoothness conditions between neighboring triangles are similar to the conditions in the one variable case. Figure 7 shows two triangles sharing an edge, and their domain points corresponding to a cubic polynomial on each triangle. These domain points are the projections of control points in $\mathbb{R}^3$, but we really think about the control points rather than the domain points. Along the common edge there are two sets of four control points each, one from each triangle. The two cubics will join continuously if the two sets coincide. The corresponding domain points are marked with red plus signs in the Figure. Since we will always require at least continuity we do not distinguish between the two sets of domain points along an edge. To obtain differentiability conditions we consider the three quadrilaterals marked in blue. The corresponding quadrilaterals of control points in $\mathbb{R}^3$ each contain four points which may or may not lie in a plane. If they do we say that the quadrilateral is *planar*. It turns out (see below) that the the two cubic polynomials will join differentiably if and only if each of the three blue quadrilaterals in R^3 is planar. They don't all have to lie in the same plane, but each of them has to be in its own plane.

The C^0 conditions for the bivariate case simply follow from he observation that along the edge of a triangle a bivariate polynomial of the form (19) reduces to a univariate polynomial of the form (3).

Deriving the quadrilateral C^1 conditions is a little more subtle than in the corresponding argument in the single variable case. However, it is instructive because it blends the algebra and geometry into a beautifully simple argument.

Directional Derivatives

To discuss differentiability we need to generalize the concept of partial derivatives. Note that the concept of partial derivatives with respect to the barycentric coordinates does not make sense (because the barycentric coordinates are not independent variables). Its indiscriminate[−2−] use leads to contradictions as in

$$1 = \frac{\partial}{\partial b_1} \underbrace{(b_1 + b_2 + b_3)}_{=1} = \frac{\partial}{\partial b_1} 1 = 0 \tag{23}$$

The appropriate generalization of the concept of a partial derivative is that of a *directional derivative.* Given a function f from R^2 to $\mathbb{R}$, a point $P \in \mathbb{R}^2$, and a direction $e \in \mathbb{R}^2$ (where the letter e suggests that the direction might be an edge of a triangle), the *derivative of f in the direction e at the point P* is defined by

$$D_e f(P) = \left(\frac{\mathrm{d}}{\mathrm{d}t} f(P + te) \right) \Big|_{t=0} . \tag{24}$$

Thus we restrict f to the line through P in the direction of e, to obtain a function of one variable t, differentiate that function in the ordinary sense, and then evaluate it at $t = 0$.

One can show easily that with ∇f denoting the gradient of f we have

$$D_e f(P) = \nabla f \cdot e \tag{25}$$

Thus if e is a standard unit vector then $D_e f$ is an ordinary partial derivative. Note that a directional derivative is well defined even if $e = 0$. In that case, $D_e f = 0$. The equation (25) means that if we know the

[−2−] Sometimes one sees an argument based on partial derivatives with respect to the barycentric coordinates that leads to a true statement. That does not mean that the argument is correct, only that the speaker is lucky.

gradient we can compute any directional derivative. It also means that if we know the directional derivatives in two linearly independent directions then we can compute the gradient.

Differentiability across an Edge

Consider now an edge shared by two triangles, and assume that the two polynomial pieces join continuously along the edge. Then a directional derivative (of any order) in the direction of the edge will also be continuous since it is determined entirely by the restriction of either of the polynomials to the edge. In order to obtain continuity of the gradient we need to have continuity of one additional derivative in a direction across the common edge.

Let p be defined by (19) and let D denote some directional derivative.

Then

$$\begin{aligned} Dp &= D \sum_{i+j+k=d} \frac{d!}{i!j!k!} c_{ijk} b_1^i b_2^j b_3^k \\ &= \sum_{i+j+k=d} \frac{d!}{i!j!k!} c_{ijk} \left(i b_1^{i-1} b_2^j b_3^k Db_1 + b_1^i j b_2^{j-1} b_3^k Db_2 + b_1^i b_2^j k b_3^{k-1} Db_3 \right) \\ &= \sum_{i+j+k=d-1} \left[\frac{d!}{(i+1)!j!k!} c_{i+1jk}(i+1)Db_1 + \frac{d!}{i!(j+1)!k!} c_{ij+1k}(j+1)Db_2 + \frac{d!}{i!j!(k+1)!} c_{ijk+1}(k+1)Db_3 \right] b_1^i b_2^j b_3^k \\ &= \sum_{i+j+k=d-1} \frac{(d-1)!}{i!j!k!} \hat{c}_{ijk} b_1^i b_2^j b_3^k \end{aligned} \tag{26}$$

where

$$\hat{c}_{ijk} = d(Db_1 c_{i+1,j,k} + Db_2 c_{i,j+1,k} + Db_3 c_{i,j,k+1}). \tag{27}$$

The derivatives Db_i are constants since the barycentric coordinates are linear function of P.

Computing Dp on each of two neighboring triangles, and comparing coefficients along the edge leads to d homogeneous equations each of which involves four control points forming the quadrilaterals in Figure 7.

The coefficients in each of those equations are the same for each quadrilateral, and independent of d after dividing by d. In particular, we get one such equation in the case $d = 1$. Now notice that the control points at the vertices of the triangle lie on the graph of the function since

$$p(V_1) = c_{d00} \qquad p(V_2) = c_{0d0} \quad \text{and} \quad p(V_3) = c_{00d}. \tag{28}$$

Thus in the case $d = 1$, the graph of p is the plane containing the three control points at the vertices. A piecewise linear function on two neighboring triangles is differentiable only if it is globally linear. Thus the four control points must lie in the same plane (which for $d = 1$ is also the graph of p). Now consider the case $d > 1$. The quadrilaterals in the domain are *similar* to the quadrilateral formed by the two triangles. Thus the same algebraic relation between the corner points of the small quadrilaterals has the same meaning as it has for the large quadrilateral formed by the two triangles: the control points at the vertices must be planar.

This gives our geometric criterion: **Two polynomials on neighboring triangles will be differentiable across the common edge if it is continuous, and each of the quadrilaterals illustrated in Figure 7 is planar.** Note that the quadrilaterals need not all lie in the same plane, but each has to lie in its own plane.

As an exercise, verify the following fact: If p is a differentiable function defined on two neighboring triangles, then the quadrilateral at an endpoint V of the common edge is contained in the tangent plane of p at V.

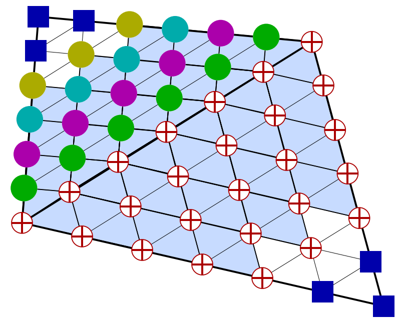

Figure 8. Higher Order Smoothness Conditions.

Higher Order Smoothness Conditions

Another way to interpret the first order smoothness conditions is as follows. Consider again one of the quadrilaterals in Figure 7. It is divided into two triangles by a segment of the common edge. The three points on one of the triangles define a linear function. The function value at the tip of the opposite triangle must be obtained by evaluating that linear function at the domain point of the tip. This interpretation carries over to differentiability conditions of order greater than 1. Consider the magenta points in Figure 8. These are the tips of larger overlapping quadrilaterals which have six points in the opposite triangle, marked with red plus signs. The C^2 conditions require that the coefficients at he magenta points be obtained by evaluating the corresponding quadratic polynomials in the opposite triangle. In general, the C^r conditions require that the coefficient be obtained by evaluating a polynomial of degree r. So the coefficients corresponding to cyan points in Figure 8 are obtained by evaluating cubic polynomials, and those corresponding to yellow points are obtained by evaluating quartic polynomials. Because of the similarity of all the triangles involved, the barycentric coordinates of the tip with respect to its individual triangle are the same as the barycentric coordinates of the corresponding vertex of the large triangle with respect to the opposite large triangle. The two polynomials in Figure 8 agree to fourth order along the common edge, and the coefficients corresponding to points marked with blue boxes do not enter any smoothness conditions.

To obtain algebraic smoothness conditions, suppose the triangle T with vertices V_1, V_2 and V_3 shares the edge e with the triangle $\hat{T}$ with vertices V_1, V_2 and V_4, and let a_1, a_2, and a_3 be the barycentric coordinates of V_4 with respect to T. Thus

$$V_4 = \sum_{i=1}^{3} a_i V_i, \qquad \sum_{i=1}^{3} a_i = 1 \tag{29}$$

Moreover, suppose that p on T is defined by (19) and that on $\hat{T}$

$$\hat{p}(P) = \sum_{i+j+k=d} \frac{d!}{i!j!k!} \hat{c}_{ijk} \hat{b}_1^i \hat{b}_2^j \hat{b}_4^k. \tag{30}$$

where $\hat{b}_1$, $\hat{b}_2$, and $\hat{b}_4$ are the barycentric coordinates of P on $\hat{T}$. Then the smoothness conditions through order r are given by

$$\hat{c}_{ijk} = \sum_{\mu+\nu+\kappa=k} \frac{k!}{\mu!\nu!\kappa!} c_{i+\mu,j+\nu,\kappa} a_1^\mu a_2^\nu a_3^\kappa, \qquad i+j+k=d, \quad k=0,1,\ldots,r. \tag{31}$$

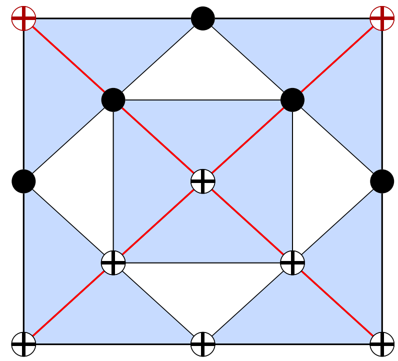

Figure 9. S_2^1 on a singular vertex.

Subdivision

Suppose we evaluate a polynomial p at a point $P = V_4$. We obtain three triangles:

$$T_1 = \{V_2, V_3, V_4\}, \quad T_2 = \{V_1, V_3, V_4\}, \quad \text{and} \quad T_3 = \{V_1, V_2, V_4\}. \tag{32}$$

On each triangle, p can be represented in Bernstein-Bézier Form with respect to that triangle. The coefficients of those representations are obtained by the de Casteljau algorithm. Specifically, they are

$$c_{0ij}^k \text{ on } T_1, \quad c_{i0j}^k \text{ on } T_2, \quad \text{and} \quad c_{ij0}^k \text{ on } T_3, \qquad \text{where} \qquad i+j+k=d. \tag{33}$$

Indeed, the point V_4 need not lie in the triangle T, and so one can use the de Casteljau algorithm to obtain a representation of p on a triangle adjoining T.

Splines

Splines are smooth piecewise polynomial functions. They can be defined on a variety of partitions of an underlying domain. In one variable this is pretty simple. We partition the interval $I = [a, b]$ into n subintervals (of arbitrary non-zero lengths) and define

$$S_d^r(I) = \{s \in C^r(I) : s|_J \in P_d \quad \text{for all subintervals } J\} \tag{34}$$

where P_d is the $d+1$-dimensional space of (univariate) polynomials of degree d, and $0 \le r \le d$. A moment's thought will show that

$$\dim S_d^r(I) = d + 1 + (n-1)(d-r). \tag{35}$$

Bivariate Splines

Two dimensional domains can be partitioned in various ways. The most interesting case is the partition of a polygonal domain into triangles whose interiors are pairwise distinct, and where the intersection of two triangles is empty, a common vertex, or a common edge. Such partitions are called *triangulations*. Let T be the triangulation, and Ω the union of all triangles in T. Then we define

$$S_d^r(\Omega) = \left\{s \in C^r(\Omega) : s|_t \in P_d^2 \quad \text{for all } t \in T\right\}. \tag{36}$$

The main difference between $S_d^r(I)$ and $S_d^r(T)$ is that in the bivariate case the dimension depends not just on the number of triangles and the way in which they are connected, but also on the precise location of the vertices. By contrast, in the univariate case the dimension of the space depends only on the number of subintervals (in addition to r and d).

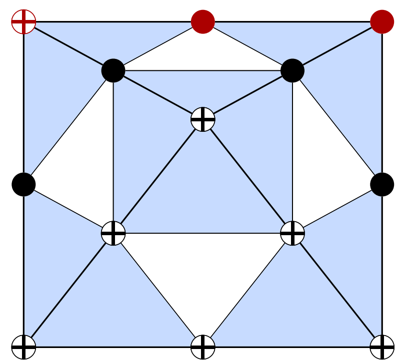

Figure 10. S_2^1 on a generic 4-star.

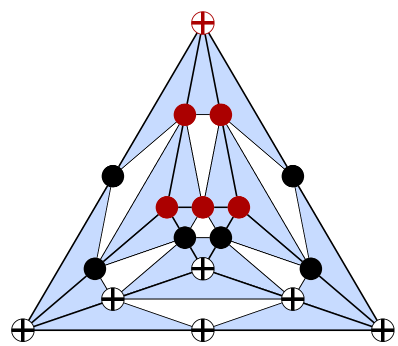

Figure 11. S_2^1 on the Morgan-Scott Split.

Figures 9 and 10 provide the simplest and best known illustration of this phenomenon. Figure 9 shows four triangles meeting at a vertex such that the four edges form two parallel pairs. Such a vertex is called *a singular vertex*. The Figure also shows the control points for S_2^1 on that triangulation. The coefficients corresponding to points marked with plus signs can be arbitrarily assigned. The coefficients at the points marked with full circles are implied by the smoothness conditions. Note that the because of the geometry of the triangulation the quadrilaterals have degenerated into triangles, and the condition that the four control points be planar actually means that the three control points at the base of the triangle are collinear in $\mathbb{R}^3$ while the coefficient at the fourth point is arbitrary. Thus the two coefficients at points marked with red plus signs do not enter any smoothness conditions and can be arbitrarily assigned.

On the other hand, if the location of the central vertex is more generic, as illustrated in Figure 10, then assigning the coefficients at six points in one triangle implies values for the values of all other coefficients. The dimension of S_2^1 is 7 on a singular vertex, but only 6 on a generic vertex shared by four triangles.

Singular vertices are not the only sources of dimension increases. Another well-known example is the Morgan-Scott split shown in Figure 11. If the points are arranged symmetrically then the dimension of S_2^1 is 7. Assigning the coefficients in one triangles leaves one coefficient (among the points shown in red) still

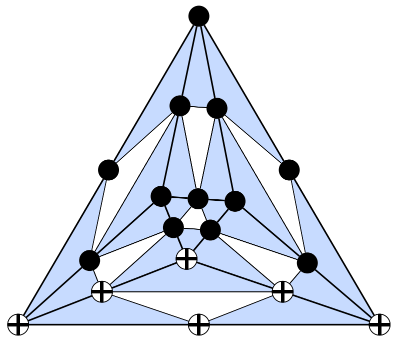

Figure 12. S_2^1 on the generic Morgan-Scott Split.

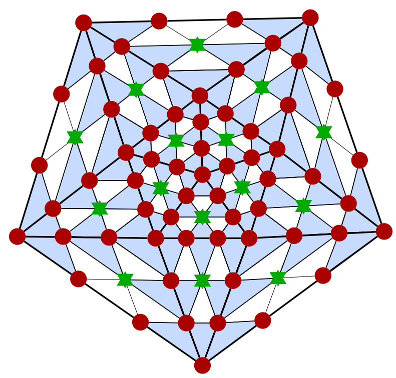

Figure 13. The dimension of $S_3^1(T)$.

available. On the other hand, if the points are in a more generic position, as shown in Figure 12, then the dimension is only 6.

Generic Dimension

It turns out that for every spline space $S_d^r(\Omega)$ there is a number D, which is such that either the dimension of $S_d^r(\Omega)$ equals D, or else there is an arbitrarily small perturbation of the location of the vertices that will cause the dimension of $S_d^r(\Omega)$ to become D. This is called the *generic dimension* of $S_d^r(\Omega)$, and it is also the smallest possible dimension.

Here is a sketch of the argument showing that the above statement in indeed true.

Think of a spline in $s \in S_d^r(\Omega)$ as a function $s \in S_d^0(\Omega)$ that satisfies additional smoothness conditions as described above. These can be expressed as a homogeneous linear system

$$Ac = 0 \tag{37}$$

where A is an $m \times n$ matrix and c is the vector of coefficients of s as a function in $S_d^0(\Omega)$. The entries of A are rational functions of the locations of the vertices of $\triangle$. The dimension of $S_d^r(\Omega)$ equals $n - R$ where R is the rank of A. Suppose the vertices of $\triangle$ are chosen such that the dimension of $S_d^r(\Omega)$ is as small as possible. That means the rank is as large as possible. That rank equals the size of a largest square submatrix of A whose determinant is non-zero. Such determinants are rational functions of the locations of the vertices of $\triangle$. A non-zero rational function cannot vanish in the neighborhood of a point. If such a determinant is in fact equal to zero for a particular set of vertex locations then there exists an arbitrarily small perturbation of those locations that will make the determinant non-zero and thus cause the rank of A to assume its largest, and the dimension of $S_d^r(\Omega)$ its smallest, value.

You might take the view, and some people do, that one should simply focus on the generic dimension and avoid spline spaces with non-generic dimensions. However, matters are not that simple. For one, it is a central fact in numerical analysis that if there is a mathematically singular situation, then in the context of inexact arithmetics nearby problems are ill conditioned. Second, many schemes, based on special triangulations of various kinds, actually exploit the additional degrees of freedom provided by non-generic dimensions.

The Generic Dimension of S_4^1

As an example of the utility of the Bernstein-Bézier Form we'll use it to show that the generic dimension of $S_4^1(\Omega)$ equals $6V-3$ where V is the number of vertices of $\triangle$.

The proof is by induction in the number of vertices. The statement is clearly true for $V=3$ since in that case the spline is just a quartic polynomial with $6V-3=15$ coefficients.

For the induction step assume the statement is true for a triangulation with $V-1$ vertices. Take a triangulation with V vertices and remove a boundary vertex and all attached triangles such that the remaining triangles still form a triangulation (with $V-1$ vertices). (Exercise: convince yourself that this is always possible).

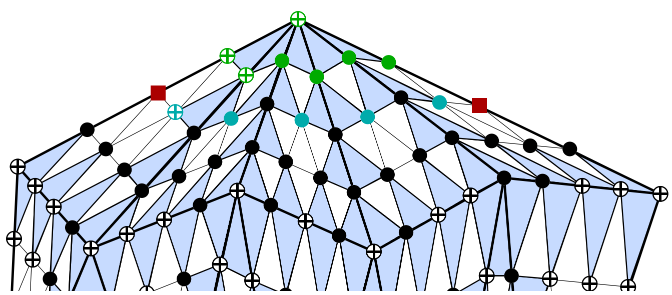

Figure 14. $\dim S_4^1 = 6V-3$.

We need to show that in this case the dimension drops by 6, or, equivalently, that adding the triangles and the vertex increases the dimension by 6.

The step is illustrated in Figure 14. We extend a spline in the existing triangulation into the five triangles shown in the Figure. The vertex marked with a green plus sign is the new vertex. Coefficients belonging to domain points marked in black are determined by the spline on the existing triangulation. Any such spline can be extended into the new triangles. The two coefficients marked with red boxes do not enter any smoothness conditions and can be assigned arbitrary values. Similarly, the three points marked with green plus signs can be assigned arbitrary values (and determine the tangent plane at the new vertex). This leaves the points marked in cyan. Any one of these can be assigned an arbitrary value (e.g., the one marked with a cyan plus sing), and doing so will imply values for the other cyan points. We assigned a total of 6 coefficients which shows that the dimension increases by 6.

As an exercise, think about how the induction could fail in the case of a non-generic dimension.

The dimension of S_3^1

The best known open question about bivariate splines concerns the dimension of $S_3^1(T)$. One can get a lower bound on $\dim S_3^1$ be following this procedure (which is illustrated in Figure 13):

1. Count 3 points for each vertex of the triangulation, representing interpolation to function and gradient values at the vertices. This corresponds to the coefficients in the *1-disks* around the vertices. Points in these disks are indicated with filled red circles in the Figure 13. The disks do not overlap, and they do not contain the central control points in each triangle, which are indicated by green stars. This gives us 3 degrees of freedom for each vertex, and one for each triangle, for a total of N, say. Smoothness conditions around the vertices, corresponding to quadrilaterals containing a vertex, are satisfied.
2. However, we also have to satisfy the central smoothness condition across each interior edge, whose quadrilateral connects two of the central control points. This leads us to subtract 1 from N, for each interior edge.
3. However, as explained above, we do gain one extra degree of freedom for each singular vertex.

Carrying out the algebra, and using the combinatoric formulas (17) for triangulations, one obtains:

$$\dim S_3^1 \geq 3V_B + 2V_1 + 1 + \sigma \tag{38}$$

where

$$\begin{aligned} V_B &= \text{number of boundary vertices of } T \\ V_B &= \text{number of interior vertices of } T \\ \sigma &= \text{number of singular vertices of } T \end{aligned} \tag{39}$$

The true dimension of S_3^1 equals the lower bound in (39) in all known cases. Most researchers in the field will conjecture that the lower bound always gives the true dimension, but nobody has been able to prove it, or come up with a counter example. The conjecture can be restated as saying that the only source of non-generic dimensions for S_3^1 are singular vertices. If you can prove an exact dimension formula for S_3^1 I would very much like to hear from you!

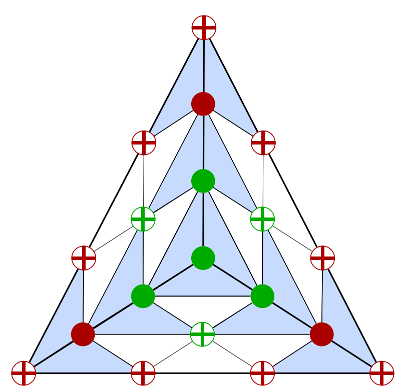

Figure 15. The Clough-Tocher Scheme.

The Clough-Tocher Element

The concepts described in these notes allow an easy interpretation of the well-known piecewise cubic C^1 Clough-Tocher Finite Element illustrated in Figure 15. Each triangle in a triangulation is divided into three micro-triangles. On each such subdivided macro triangle we interpolate to function and gradient values at the vertices. This implies the values of the coefficients indicated with red plus signs. The coefficients indicated with red full circles are implied by the smoothness conditions. We also interpolate to a perpendicular cross-boundary derivatives at the midpoint of each edge. This implies the values of the coefficients indicated with

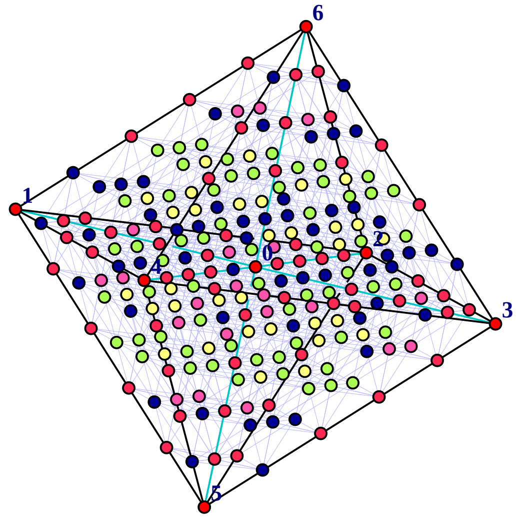

Figure 16. Trivariate Splines.

green plus sings. The remaining coefficients indicated with full green circles are implied by the smoothness conditions. An analysis in terms of the cartesian form (15) of bivariate polynomials would be much more complicated.

The concepts briefly touched upon in these notes apply in any spatial dimension. Figure 16 shows the domain points for a piecewise quintic trivariate spline defined on a partition of an octahedron into 8 tetrahedra. The colors in that picture do carry meaning, and to appreciate the Figure you need to be able to rotate the octahedron and see it in three dimensions. You can do that on the web page indicated below.

More Information

This has been a very quick sightseeing trip in the area of multivariate splines and their representation in B-form. For more information consult the comprehensive monograph "Spline Functions on Triangulations" by Ming-Jun Lai and Larry L. Schumaker (Cambridge University Press, 2007, ISBN 0521875927) or visit the author's web pages

http://www.math.utah.edu/~pa/MDS and http://www.math.utah.edu/~pa/3DMDS