

# MA 323 Geometric Modelling

## Course Notes: Day 29

### Composite Rectangular Surface Patches

David L. Finn

Today, we want to extend rectangular Bezier patches to composite Bezier patches. This is the natural extension of the methods from  $C^k$  Bezier splines and  $G^k$  Bezier splines to patches. For simplicity, we will only consider  $C^k$  smoothness the ability to join patches in a  $G^k$  manner can also be done in the natural manner. The ability to join patches together relies upon some different concept as to how the patches are to be joined together. This is the start of adding a topological condition into our methods. This did not really play a role in curve methods because there is really only one manner of joining two curves. In joining surface patches, one must consider how the patches are joined, for instance given two rectangular Bezier patches  $m \times n$  and  $p \times q$ , how is it possible to join them together? Not unless one of the following hold  $m = p$ ,  $m = q$ ,  $n = p$  or  $n = q$  so that the common Bezier curve on an edge can be the same curve. As a further example, given three rectangular patches, is it possible to join them together smoothly as illustrated in the diagram below?

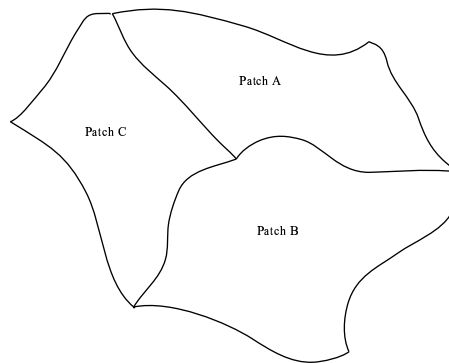


Figure 1: Can three rectangular patches be joined together in this manner?

#### 29.1 Composite patches

To create a surface, by joining individual patches, one method is to make sure the surfaces have an entire curve in common otherwise one patch will pass through another patch or the patches will not meet. Joining to patches continuously is pretty easy as long as the patches are the same order. It is only necessary that they share a common edge set of control points, see figure below for continuously joining two bicubic patches.

It is important that the size of the control grids be compatible to join the patches continuously. It is also important that we are considering control points. If we use data points

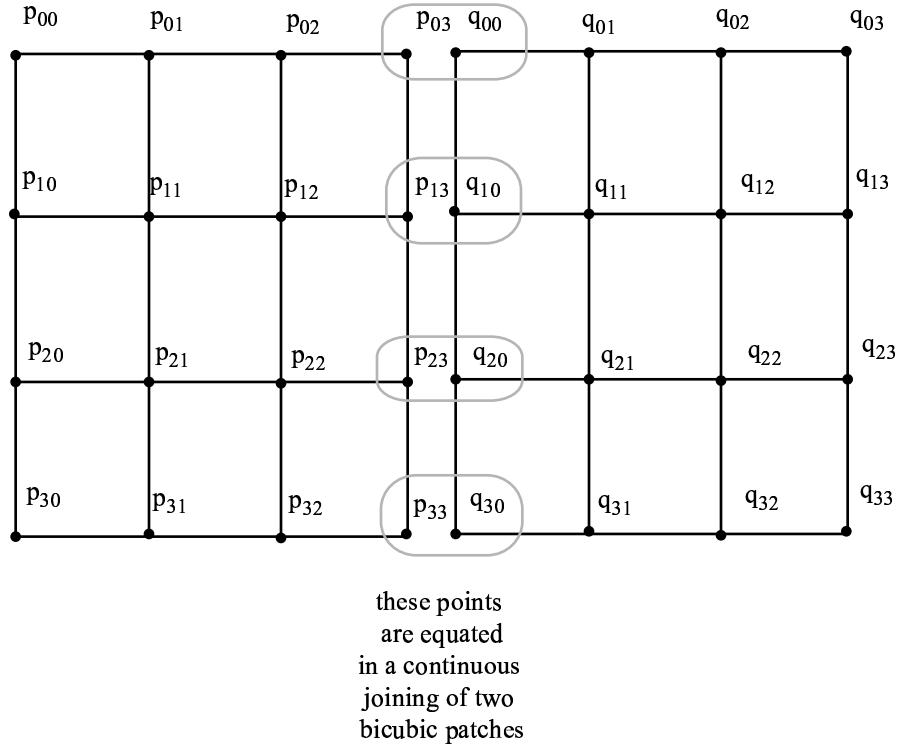


Figure 2: Joining two bicubic Bezier patches continuously

and interpolate or approximate instead, we are not guaranteed that the patches will join continuously. It can be forced by using the “right” values for the parameters.

The basic question to consider is: Why does sharing the same control point ensure that the patches are joined continuously? The answer lies in the construction of the patch. Recall a patch is defined by

$$X(u, v) = \sum_{i=0}^m \sum_{j=0}^n B_i^m(u) B_j^n(v) p_{ij}.$$

The edge control points in the control grid correspond to  $u = 0$ ,  $u = 1$ ,  $v = 0$ ,  $v = 1$ . We will consider the edges  $u = 0$  and  $u = 1$ . When  $u = 0$ ,  $B_i^m(0) = 0$  if  $i > 0$  and  $B_0^m(0) = 1$ . Therefore, the edge  $X(0, v)$  of the patch corresponds to the Bezier curve

$$\sum_{j=0}^n B_j^n(v) p_{0,j}.$$

Similarly, when  $u = 1$ ,  $B_i^m(1) = 0$  if  $i < m$  and  $B_m^m(1) = 1$ , so the edge of the patch  $P(1, v)$  corresponds to the Bezier curve

$$\sum_{j=0}^n B_j^n(v) p_{m,j}.$$

Thus, given two patches with control points  $p_{i,j}$  and  $q_{i,j}$  as shown below with  $p_{m,j} = q_{0,j}$ ,

we have  $X_1(1, v) = X_2(0, v)$  where  $X_1$  and  $X_2$  are the patches corresponding to control grids  $p$  and  $q$ .

## 29.2 Derivatives of Bezier patches

To join patches together smoothly, we need to examine the derivatives of patches. Given a control net for a Bezier patch (a grid of control points)  $p_{i,j}$ . The Bezier patch is given by

$$P(u, v) = \sum_{i=0}^m \sum_{j=0}^n B_i^m(u) B_j^n(v) p_{i,j}.$$

We can take the partial derivatives with respect to  $u$  and  $v$  to get

$$\frac{\partial P}{\partial u} = \sum_{i=0}^m \sum_{j=0}^n (B_i^m(u))' B_j^n(v) p_{i,j}$$

and

$$\frac{\partial P}{\partial v} = \sum_{i=0}^m \sum_{j=0}^n B_i^m(u) (B_j^n(v))' p_{i,j}.$$

Recalling the relations between the derivatives of the Bernstein polynomials,

$$\begin{aligned} \frac{d}{dt} (B_i^n(t)) &= \binom{n}{i} (i t^{i-1} (1-t)^{n-i} - (n-i) t^i (1-t)^{n-i-1}) \\ &= n \binom{n-1}{i-1} t^{i-1} (1-t)^{n-i} - n \binom{n-1}{i} t^i (1-t)^{n-1-i} \\ &= n [B_{i-1}^{n-1}(t) - B_i^{n-1}(t)], \end{aligned}$$

we have

$$\frac{\partial P}{\partial u} = \sum_{i=0}^m \sum_{j=0}^n m [B_{i-1}^{m-1}(u) - B_i^{m-1}(u)] B_j^n(v) p_{i,j}$$

and

$$\frac{\partial P}{\partial v} = \sum_{i=0}^m \sum_{j=0}^n n B_i^m(u) [B_{j-1}^{n-1}(v) - B_j^{n-1}(v)] p_{i,j}.$$

This gives a formal method for differentiating a Bezier patch from the functional representation of the Bezier patch.

We can express these forms of the derivatives in terms of first differences of the control net. We now have two first differences, one for each direction in the grid. We have a first difference in the  $u$ -direction (the  $i$  index)

$$\Delta_{1,0} p_{i,j} = p_{i+1,j} - p_{i,j},$$

and a first difference in the  $v$ -direction (the  $j$  index)

$$\Delta_{0,1} p_{i,j} = p_{i,j+1} - p_{i,j}.$$

By rearranging the sums, we have

$$\frac{\partial P}{\partial u} = \sum_{i=0}^{m-1} \sum_{j=0}^n m B_i^{m-1}(u) B_j^n(v) \Delta_{1,0} p_{i,j}$$

and

$$\frac{\partial P}{\partial v} = \sum_{i=0}^m \sum_{j=0}^n n B_i^m(u) B_j^{n-1}(v) \Delta_{0,1} p_{i,j}.$$

Notice that each derivative is also a Bezier patch. The grid on which the Bezier patch is defined is smaller and consists of vectors.

To take the second derivatives of a Bezier patch, we proceed in a similar manner. As with Bezier curves, since the derivative of a Bezier patch is a Bezier patch, we can use the above methods to define the second derivatives. The second derivatives are

$$\frac{\partial^2 X}{\partial u^2} = \sum_{i=0}^{m-2} \sum_{j=0}^n m(m-1) B_i^{m-2}(u) B_j^n(v) \Delta_{2,0} p_{i,j}$$

and

$$\frac{\partial^2 X}{\partial u \partial v} = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} mn B_i^{m-1}(u) B_j^{n-1}(v) \Delta_{1,1} p_{i,j}$$

and

$$\frac{\partial^2 X}{\partial v^2} = \sum_{i=0}^m \sum_{j=0}^{n-2} n(n-1) B_i^m(u) B_j^{n-2}(v) \Delta_{0,2} p_{i,j}$$

where  $\Delta_{k,l} p_{i,j}$  are the appropriate differences. The differences above are given by

$$\begin{aligned} \Delta_{2,0} p_{i,j} &= \Delta_{1,0} p_{i+1,j} - \Delta_{1,0} p_{i,j} \\ &= p_{i+2,j} - 2p_{i+1,j} + p_{i,j} \end{aligned}$$

and

$$\begin{aligned} \Delta_{1,1} p_{i,j} &= \Delta_{1,0} p_{i,j+1} - \Delta_{1,0} p_{i,j} \\ &= \Delta_{0,1} p_{i+1,j} - \Delta_{0,1} p_{i,j} \\ &= p_{i+1,j+1} - p_{i,j+1} - p_{i+1,j} + p_{i,j} \end{aligned}$$

and

$$\begin{aligned} \Delta_{0,2} p_{i,j} &= \Delta_{0,1} p_{i,j+1} - \Delta_{0,1} p_{i,j} \\ &= p_{i,j+2} - 2p_{i,j+1} + p_{i,j}. \end{aligned}$$

We can proceed to take higher derivatives in a similar manner.

The mixed partial derivative  $\partial^2 X / \partial u \partial v$  has special significance for patches, as of the second partial derivatives at corners it is the only one that uses the interior control points. This vector ( $\partial^2 X / \partial u \partial v$ ), all derivatives of patches are vectors, is called the twist vector for it states how the tangent vectors  $\partial X / \partial u$  and  $\partial X / \partial v$  twist as the other variable is moved. We will explore this in detail tomorrow.

### 29.3 Joining Patches Smoothly

Joining patches together smoothly is now easy. It is equivalent to arranging the control grids of the two patches so that each line forms a  $C^k$  spline. For instance, consider the control grids  $p_{ij}$  and  $q_{ij}$  below with  $p_{mj} = q_{0j}$  so they join continuously.

To have the patches joined in a  $C^1$  fashion, we need

$$p_{mj} = q_{0j} = \frac{1}{2}(p_{m-1,j} + q_{1,j})$$

for all  $j$ . Likewise to join two patches in a  $C^2$  fashion, we need in addition to the conditions for  $C^0$  and  $C^1$  joining that

$$p_{m-2,j} - 2p_{m-1,j} + p_{m,j} = q_{2,j} - 2q_{1,j} + q_{0,j}$$

for all  $j$ .

Constructing  $C^1$  patches and  $C^2$  patches (using bicubic patches as the base patch) is now easy. We just need to set up the control points in a  $C^k$  fashion, that is supply the spline control points in the correct grid structure, in both directions for the grid. Here, we are aided directly by the patch construction as a tensor product surface. Individual rows (or columns) in the spline control grid will correspond to constructing the corresponding Bezier splines. See diagram below for the structure to construct a  $C^1$  joining of four patches in a rectangular patchwork fashion

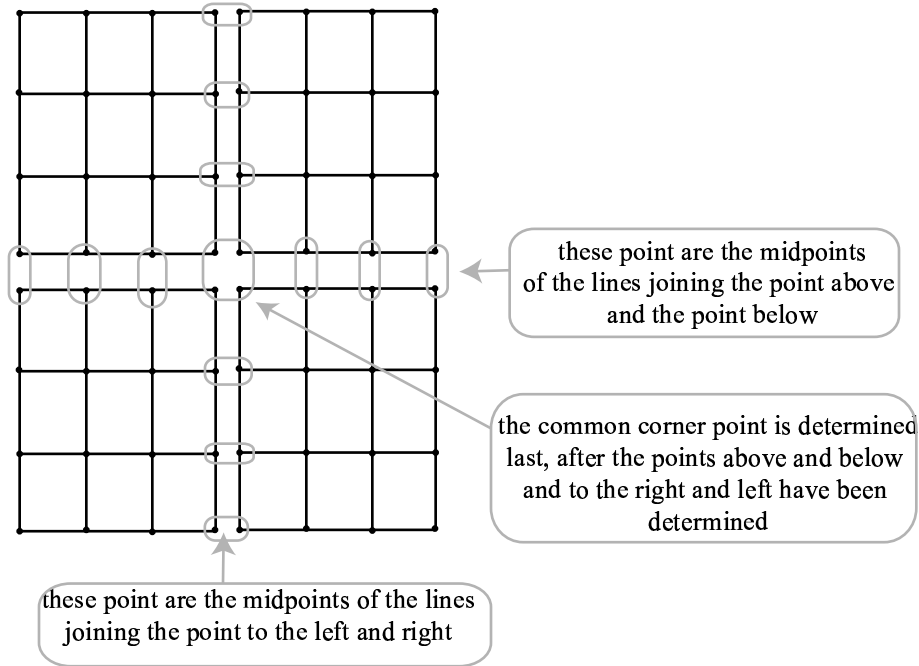


Figure 3: Joining of four rectangular patches in a  $C^1$  fashion.

We can also extend the Bezier patches to B-spline patches by supplying knot sequences, and using the knot sequences with a control grid to define a tensor product B-spline patch. We use the knot sequence in the  $u$  direction to define basis functions  $N_{i,k}(u)$  and a knot sequence in the  $v$  direction to define basis functions  $N_{j,l}(v)$ . A B-spline patch is then

$$p(u, v) = \sum_i \sum_j N_{i,k}(u) N_{j,l}(v) p_{i,j}.$$

As B-splines include Bezier curves and Bezier splines, B-spline patches include Bezier patches and composite Bezier patches. All that is needed is to choose the right type of knot sequence.

## 29.4 Other Ways of Joining Patches

Let us return to the question at the start of the notes, can you join three rectangular patches together in the following manner

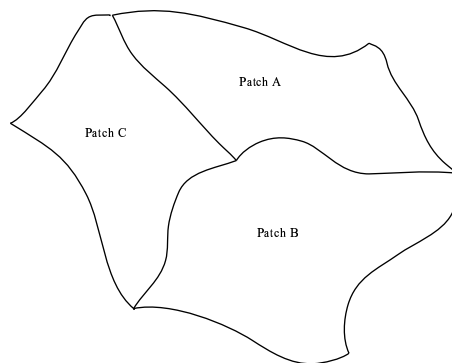


Figure 4: Can three rectangular patches be joined continuously in this manner?

The answer is yes. One must just have the requisite placements of control points correct. Functionally the patch can not be represented in a nice global coordinate system, as the patchwork quilt nature of joining in the previous subsection suggests. For instance, here is the compatibility constraints for a  $C^0$  joining.

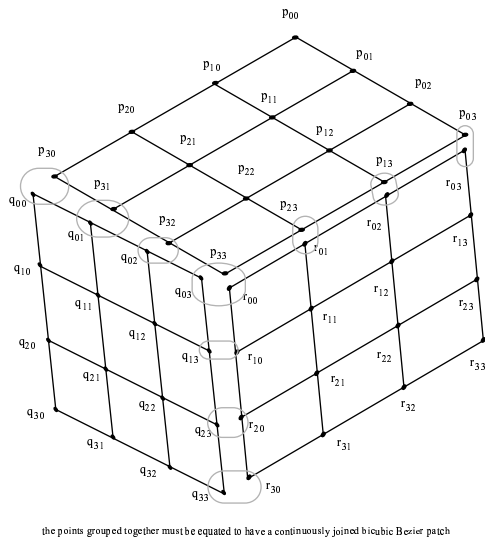


Figure 5: Compatibility conditions for a continuous joining

A continuous joining in this fashion is pretty straight forward, but what about a differentiable joining? Does it even make sense to talk about a  $C^1$  joining of rectangular patches in this manner? In fact, a  $C^1$  joining can be considered across each edge, but this does not mean anything in terms of a global differentiability condition in a variable defined on the surface. This is really the notion of a  $G^1$  surface, meaning that there is a well-defined tangent plane everywhere. Constructing the surface so that the joint across edge is  $C^1$  will ensure a well-defined tangent plane. However, this places extra compatibility conditions

on the three interior points  $p_{22}$ ,  $q_{12}$  and  $r_{11}$  surrounding the common corner point to the rectangular patches  $p_{33} = q_{03} = r_{00}$  in the figure above. In fact, if the patches are joined across the common edges in a  $C^1$  fashion one must have that  $p_{22}$ ,  $q_{12}$  and  $r_{11}$  are coplanar. Showing this is an exercise.

Typically to create a surface out of rectangular patches in this manner requires a irregular rectangular grid, to create closed surfaces. In this situation, one does not create a patch from generalizing Bezier splines but rather looking at individual Bezier patches and joining them together in a smooth fashion.

## EXERCISES

1. Use the files **splinepatch.mws** and **splinepatch.txt** and the applets to explore the shape of rectangular  $C^1$  and  $C^2$  Bezier spline patches having open ends (a generic open surface), see diagram below. What advantage is this over Bezier patches?
2. Use the files **bsplinepatch.mws** and **bsplinepatch.txt** to explore the shape of B-spline patches.
3. Consider the figure below where two rectangular patches are joined along two common edges at a corner, see diagram below. What conditions must be placed on the patches so they join continuously (size of the patches)? so they join smoothly ( $C^1$ ) (relationships between control points)?

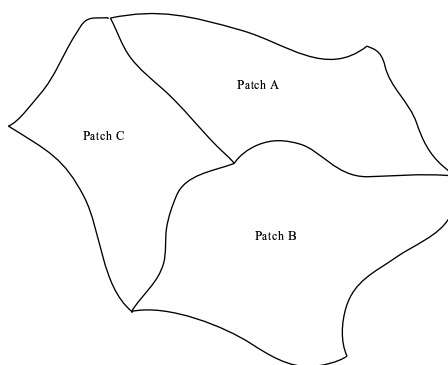


Figure 6: Can two rectangular patches be joined in this manner?

4. Show that the compatibility conditions for joining of three bicubic patches as in the diagram below in a  $G^1$  fashion, by joining the patches in a  $C^1$  manner across identified edges, closest interior points to the common corner to be coplanar.
5. Some question about joining patches together to create a closed surface,
  - (a) Is it possible to create a sphere by joining rectangular patches together in a regular rectangular grid fashion (four rectangles meet at one common point)? If so how?
  - (b) Can you do it with a irregular rectangular grid (any number of rectangles can meet at a common point)? If so how?
  - (c) What is the minimum number of rectangles needed in either case?

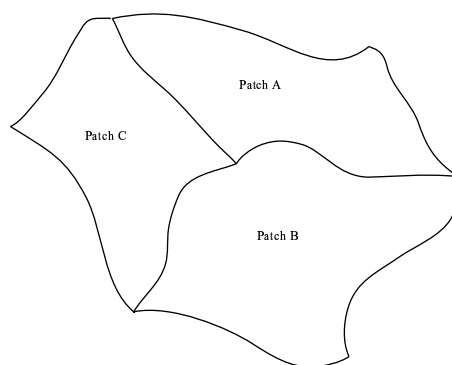


Figure 7: Can three rectangular patches be joined continuously in this manner?