

On-Line Geometric Modeling Notes

VERTEX POINTS AND EDGE POINTS

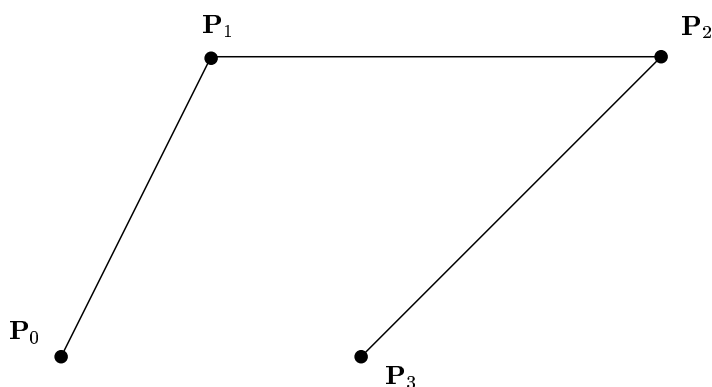
Kenneth I. Joy
Visualization and Graphics Research Group
Department of Computer Science
University of California, Davis

Overview

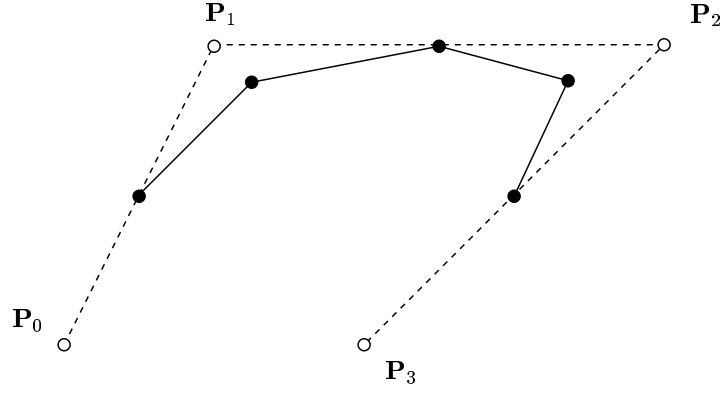
The refinement process defined by the cubic uniform B-spline curve generates a sequence of control polygons that converges to the curve. This process can be specified by examining the binary subdivision methods for the curve to obtain a unique set of control points that generates the subdivision. However, in the case of the cubic curve, there is another way to look at the generation of the points of a refined control polygon. Each of these points can be classified as either a “vertex point” or an “edge point,” and methods can be specified to calculate each point. This procedure turns out to be quite useful as they will allow us to directly calculate points on the limit curve without going through the refinement.

Classifying the Refinement Points

Suppose we are given a control polygon $\{P_0, P_1, P_2, P_3\}$.



If we use the refinement method motivated by the binary subdivision of the curve, we generate a new control polygon shown below



We note that three of the new control points appear to lie at the midpoints of the three respective line segments. These points will be classified as “edge points”. The other points all lie close to on of the vertices of the original control polygon, and will be called “vertex points”.

In this light, if we denote the new control polygon generated by the binary subdivision method as $\{\mathbf{E}_0, \mathbf{V}_0, \mathbf{E}_1, \mathbf{V}_1, \mathbf{E}_2\}$ then by combining and applying the splitting matrices that generate the binary subdivision of the curve we obtain

$$\begin{bmatrix} \mathbf{E}_0 \\ \mathbf{V}_0 \\ \mathbf{E}_1 \\ \mathbf{V}_1 \\ \mathbf{E}_2 \end{bmatrix} = \frac{1}{8} \begin{bmatrix} 4 & 4 & 0 & 0 \\ 1 & 6 & 1 & 0 \\ 0 & 4 & 4 & 0 \\ 0 & 1 & 6 & 1 \\ 0 & 0 & 4 & 4 \end{bmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix}$$

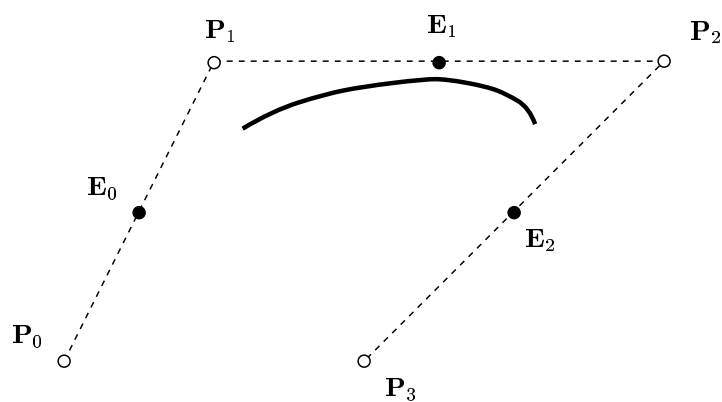
and so the edge points \mathbf{E}_0 , \mathbf{E}_1 and \mathbf{E}_2 are calculated by

$$\begin{aligned}\mathbf{E}_0 &= \frac{1}{8}(4\mathbf{P}_0 + 4\mathbf{P}_1) \\ &= \frac{\mathbf{P}_0 + \mathbf{P}_1}{2}\end{aligned}$$

$$\begin{aligned}\mathbf{E}_1 &= \frac{1}{8}(4\mathbf{P}_1 + 4\mathbf{P}_2) \\ &= \frac{\mathbf{P}_1 + \mathbf{P}_2}{2}\end{aligned}$$

$$\begin{aligned}\mathbf{E}_2 &= \frac{1}{8}(4\mathbf{P}_2 + 4\mathbf{P}_3) \\ &= \frac{\mathbf{P}_2 + \mathbf{P}_3}{2}\end{aligned}$$

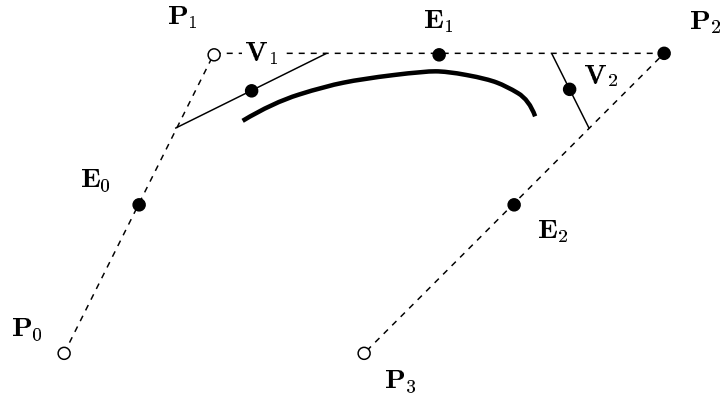
and indeed are the midpoints of the line segments connecting the original control points. These points are illustrated in the following figure:



The \mathbf{V}_0 and \mathbf{V}_1 are calculated by

$$\begin{aligned}
\mathbf{V}_0 &= \frac{1}{8}(\mathbf{P}_0 + 6\mathbf{P}_1 + \mathbf{P}_2) \\
&= \frac{1}{8}((\mathbf{P}_0 + \mathbf{P}_1) + 4\mathbf{P}_1 + (\mathbf{P}_1 + \mathbf{P}_2)) \\
&= \frac{1}{4}\left(\frac{\mathbf{P}_0 + \mathbf{P}_1}{2} + 2\mathbf{P}_1 + \frac{\mathbf{P}_1 + \mathbf{P}_2}{2}\right) \\
&= \frac{1}{4}(\mathbf{E}_0 + 2\mathbf{P}_1 + \mathbf{E}_1) \\
&= \frac{\frac{\mathbf{E}_0 + \mathbf{P}_1}{2} + \frac{\mathbf{P}_1 + \mathbf{E}_1}{2}}{2}
\end{aligned}$$

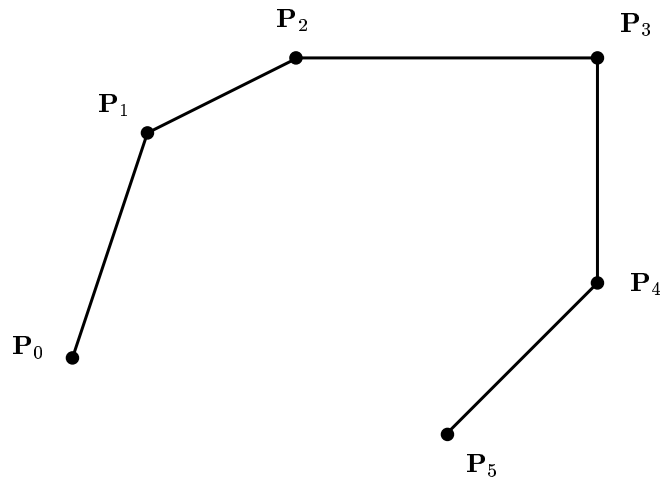
$$\begin{aligned}
\mathbf{V}_1 &= \frac{1}{8}(\mathbf{P}_1 + 6\mathbf{P}_2 + \mathbf{P}_3) \\
&= \frac{1}{8}((\mathbf{P}_1 + \mathbf{P}_2) + 4\mathbf{P}_2 + (\mathbf{P}_2 + \mathbf{P}_3)) \\
&= \frac{1}{4}\left(\frac{\mathbf{P}_1 + \mathbf{P}_2}{2} + 2\mathbf{P}_2 + \frac{\mathbf{P}_2 + \mathbf{P}_3}{2}\right) \\
&= \frac{1}{4}(\mathbf{E}_1 + 2\mathbf{P}_2 + \mathbf{E}_2) \\
&= \frac{\frac{\mathbf{E}_1 + \mathbf{P}_2}{2} + \frac{\mathbf{P}_2 + \mathbf{E}_2}{2}}{2}
\end{aligned}$$



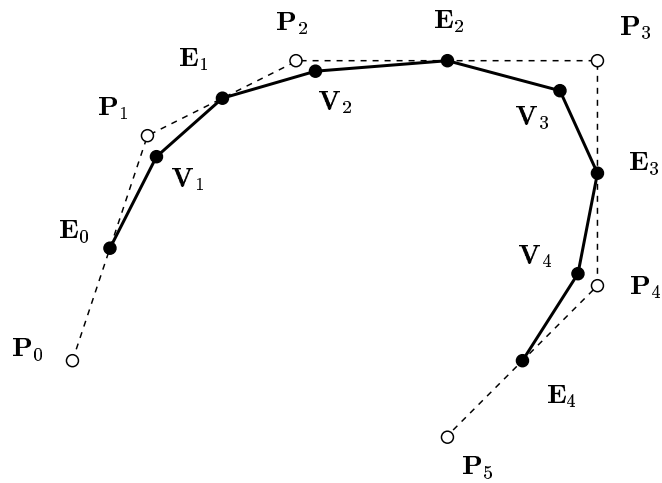
and it is seen that these are the midpoint of the line segment that joins the respective midpoints of the two line segments from the vertex point to the respective edge points. Therefore, we only need to be able to specify midpoints of line segments to be able to specify the refined control polygon.

An Example of the Refinement Algorithm

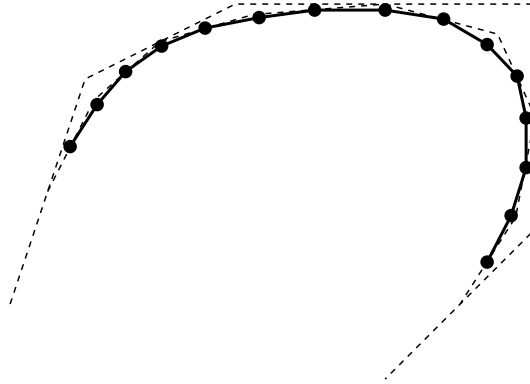
We illustrate the workings of this refinement algorithm via the following three figures. First consider a control polygon as is shown below.



Calculate the edge and vertex points producing a refinement of the original control polygon.



Assemble these points into a new control polygon and utilize it as input again to the refinement process – producing new edge and vertex points from this set of points.



The general idea behind *subdivision curves* is to utilize the control points generated through refinement as input to another refinement operation, and then continue this process until a refinement is reached that accurately represents the curve to a desired resolution.

It is well known in the case of uniform B-spline curves that these refinement points converge in the limit to the curve itself.

Summary

In the case of cubic uniform B-spline curves we can consider the refinement procedure to consist of the generation of edge points and vertex points from a given set of control points – each of which can be generated by taking only the midpoints of line segments. This classification of the control points will become very useful in the direct calculation of points on the limit curve.

References

- [1] CATMULL, E., AND CLARK, J. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-Aided Design* 10 (Sept. 1978), 350–355.