Normals of subdivision surfaces and their control polyhedra

I. Ginkel, [a,1] J. Peters [b,2] and G. Umlauf [a,1]

[a] *University of Kaiserslautern, Germany*

[b] *University of Florida, Gainesville, FL, USA*

Abstract

For planar spline curves and bivariate box-spline functions, the cone of normals of a polynomial spline piece is enclosed by the cone of normals of its spline control polyhedron. This note collects some concrete examples to show that this is not true for subdivision surfaces, both at extraordinary points and in the regular, box-spline setting. A larger set, the cross products of families of control net edges, has to be considered.

Key words: normals, splines, subdivision, control polyhedron

1 Introduction

A key property of subdivision surfaces for applications such as rendering on the computer, is that they are well approximated by their *control polyhedron*. It is therefore natural to ask whether the surface normal is well-approximated by the facet normals of the control polyhedron. Investigating this question is practically relevant since an affirmative answer to the question would allow substituting a simpler computation of bounds on the control polyhedron for complex, exact bounds on the corresponding nonlinear surface. This note records some simple examples and proofs concerning normals and control nets of box-spline and Loop subdivision surfaces. While, for planar curves and bivariate box-spline *functions*, the cone of normal directions of a polynomial spline segment is enclosed in the cone of normals of the spline control polyhedron [1], Section 2 shows that such simple tight bounds do not exist for

[1] `ginkel|umlauf@informatik.uni-kl.de`
[2] `jorg@cise.ufl.edu`

box-spline and subdivision *surfaces*. Figure 4 gives examples of planar control polyhedra where the orientation of some facets flip under refinement resulting in self-overlap that was not present in the initial control polyhedron. Section 3 confirms, however, that a larger set of cross products of control net edges yields bounds on the normals of box-spline surfaces and at irregular points of Loop subdivision.

2 Normal cones of control facets do not bound

A simple example shows that the cone of normal directions $\mathbf{n}(\mathbf{p}, \mathsf{u})$, $\mathsf{u} \in U$, of a parametric polynomial box-spline surface piece (patch) $\mathbf{p}$ mapping to $\mathbb{R}^3$ is in general not in the cone of the normals of its control facets. A variant of the argument shows that, in general, no subdivision scheme for computing normals exists that leaves linear subspaces invariant. In particular, there is no linear averaging scheme on the facet normals that generates the limit normals.

First, we consider the three-direction box spline with direction matrix $\Xi := \left[\begin{smallmatrix} 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{smallmatrix}\right]$. The control facets of this box spline are triangles. Already in [1] it was shown that the normal of a polynomial piece $\mathbf{p}$ of a box-spline surface with averaging matrix Ξ is in general not in the normal cone spanned by its control facet normals. Figure 1 complements the example in [1] to show that the claim does not hold for valences greater than six under Loop subdivision [2].

Lemma 1 *The limit normal of a Loop subdivision surface at an extraordinary point does not, in general, lie in the cone spanned by the control facet normals.*

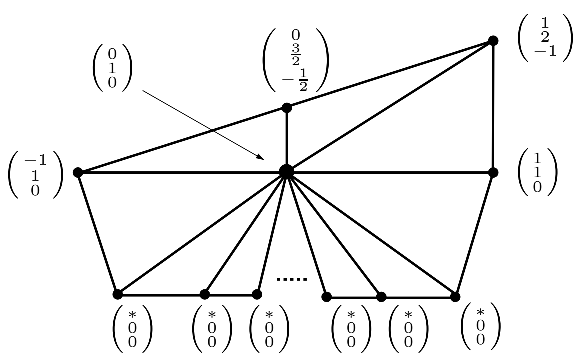

Fig. 1. Mesh neighborhood of a vertex of high valence.

To verify the claim, recall that a control net N with a single irregularity of order n can be subdivided by applying a square subdivision matrix. For the algorithm of Loop this matrix is diagonalizable. Thus, N can be expressed in terms of right-eigenvectors $\mathbf{v}_i$ of the subdivision matrix:

$$N = \sum \mathbf{v}_i \mathbf{d}_i, \ \mathbf{d}_i \in \mathbb{R}^3.$$

The tangent plane at an extraordinary point $\mathbf{c}_\infty$ of a C^1 subdivision surface is spanned by $\mathbf{d}_1$ and $\mathbf{d}_2$. These two vectors can be computed by scaling the neighbor control points $\mathbf{p}_i$ of the irregularity with weights and summing. The weights are given by the stencils in Figure 2 where $c_i := \cos(\frac{2\pi i}{n})$, $s_i := \sin(\frac{2\pi i}{n})$ for $i = 0, \ldots, n-1$ (see e.g. [3]), i.e. $\mathbf{d}_1 = \sum c_i \mathbf{p}_i$.

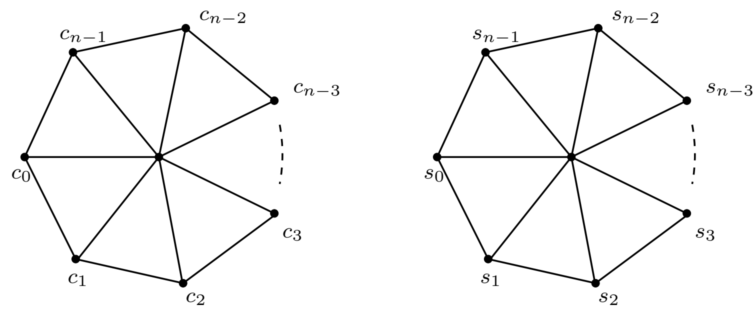

Fig. 2. Stencils for the calculation of the tangent directions for Loop surfaces.

Proof Consider Figure 1. All but four points are placed on the x-axis and so that all triangles connecting the x-axis to the central, extraordinary point are nondegenerate. Evidently, regardless of the choice of the spacing on the x-axis, the facet normal of these triangles have a zero x-coordinate. The remaining three triangles are coplanar since $\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$ and $\begin{pmatrix} 0 \\ 3 \\ -1 \end{pmatrix}/2$ are midpoints of edges and the normal to the plane of the three triangles has a zero x-component. It suffices to show that the limit normal has a nonzero x-component. We apply the stencils of Figure 2 to the net in Figure 1 and get

$$\mathbf{d}_1 = \begin{pmatrix} * \\ 1+\frac{3}{2}c_1+2c_2+c_3 \\ -\frac{1}{2}c_1-c_2 \end{pmatrix}, \mathbf{d}_2 = \begin{pmatrix} * \\ -\frac{3}{2}s_1-2s_2-s_3 \\ \frac{1}{2}s_1+s_2 \end{pmatrix}.$$

For valence $n > 4$, the x-coordinate of the (unnormalized) limit normal

$$\mathbf{n} = \begin{pmatrix} c_1 s_1 \\ * \\ * \end{pmatrix}$$

is strictly positive. $\square$

The flaw in the statement in the Introduction is that the normal is bilinear with respect to the action of the subdivision matrices and hence, we should not expect to find a linear averaging scheme for normals. The following lemma confirms that, in any case, normals do not stay confined to a subpace; so some facet normals of the refined mesh can not be written as the average of the input normals.

Lemma 2 *No subdivision scheme that leaves Euclidian subspaces invariant can generate the facet normals of the* Ξ*-subdivided mesh from the facet normals of the input mesh.*

Proof We perform one subdivision step using the box-spline subdivision rules (see e.g. [4]) on the data of Figure 3, *left*, resulting in the control net shown in Figure 3, *right*. We consider the facet normals at the apex of the central triangle. It has normal direction

$$\left(\begin{pmatrix}32\\0\\-4\end{pmatrix}-\begin{pmatrix}1\\-1\\-3\end{pmatrix}\right)\times\left(\begin{pmatrix}16\\-16\\0\end{pmatrix}-\begin{pmatrix}1\\-1\\-3\end{pmatrix}\right)=\begin{pmatrix}31\\1\\-1\end{pmatrix}\times\begin{pmatrix}15\\-15\\3\end{pmatrix}=\begin{pmatrix}-12\\ *\\ *\end{pmatrix}.$$

Since the normal direction has a nonzero x-component, it is not in the space spanned by the facet normals of the original control mesh. □

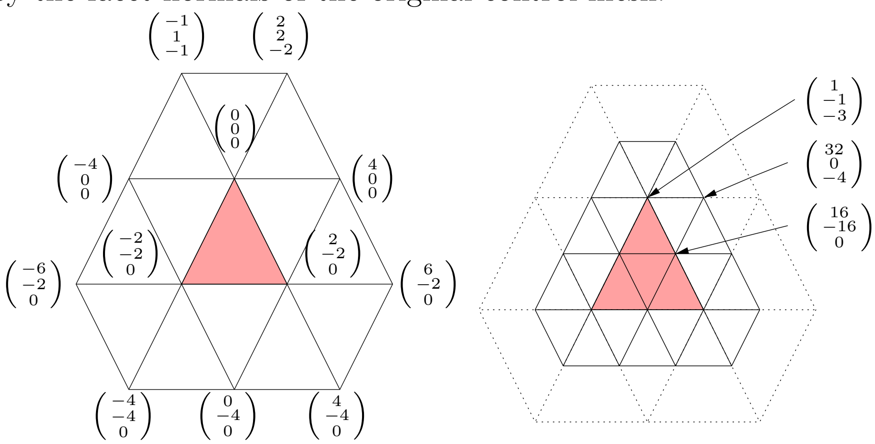

Fig. 3. (*left*)Diagram of a three-direction box-spline control net. Only the top two coefficients have a non-zero z-component. The (x, y) coordinates of all coefficients are evenly distributed, except for the top left which is chosen (for simplicity) as the average of the top right and $(-4, 0, 0)^T$. (*right*) Coefficients (scaled by 16) of the once-refined control net.

A variant of the running example, the 4×4 control net

$$\begin{matrix}\begin{pmatrix}-2\\0\\-1\end{pmatrix}&\begin{pmatrix}-1\\0\\-1\end{pmatrix}&\begin{pmatrix}0\\0\\-1\end{pmatrix}&\begin{pmatrix}1\\0\\-2\end{pmatrix}\\ \begin{pmatrix}-2\\0\\0\end{pmatrix}&\begin{pmatrix}-1\\0\\0\end{pmatrix}&\begin{pmatrix}0\\0\\0\end{pmatrix}&\begin{pmatrix}1\\0\\0\end{pmatrix}\\ \begin{pmatrix}-2\\-1\\0\end{pmatrix}&\begin{pmatrix}-1\\-1\\0\end{pmatrix}&\begin{pmatrix}0\\-1\\0\end{pmatrix}&\begin{pmatrix}1\\-1\\0\end{pmatrix}\\ \begin{pmatrix}-2\\-2\\0\end{pmatrix}&\begin{pmatrix}-1\\-2\\0\end{pmatrix}&\begin{pmatrix}0\\-2\\0\end{pmatrix}&\begin{pmatrix}1\\-2\\0\end{pmatrix}\end{matrix}$$

shows that reducing the number of box-spline averaging directions does not remedy the situation: the normals of a polynomial piece of a *bi-cubic tensor-product spline* also do not lie in the cone generated by the bilinear facets since the normal at the upper right vertex has a nonzero x-component. (The control points can be visualized as evenly spread out on the plane, followed by flipping down the topmost row and pulling the rightmost point a bit further down.) The same regular grid, interpreted as the 16 control points of a *bi-cubic patch in Bernstein-Bézier form*, yields a normal at parameters $(1, .5)$, the middle of the right boundary, that has a non-zero x-component.

3 Normal cones of control edges bound

A larger set of cross products generates a cone that encloses the limit normals of the patch. Let e_1 and e_2 be two independent averaging directions of a smooth box spline and denote by I the set of indices of control points in the support of the normals in question. Then the set of cross products of *all* edge vectors of the control net in the direction e_1 with *all* edge vectors in the direction e_2 yields useful bounds.

Lemma 3 *The normal of a polynomial piece* $\mathbf{p}$ *of a bivariate* C^1 *parametric box-spline surface with independent averaging directions* e_1 *and* e_2 *is in the normal cone spanned by the cross products of each difference in the direction* e_1 *with each difference in the direction* e_2 *of control points* $\mathbf{p}_\mathsf{i}$, $\mathsf{i} \in \mathsf{I}$.

Proof Since the cross product results in a polynomial of degree $2d-2$ and the box-spline functions b_i form a non-negative partition of 1, we can bound the normal directions by

$$\begin{aligned}\mathbf{n}(\mathbf{p},\mathsf{u}) := \frac{\partial \mathbf{p}}{\partial \mathsf{u}_1} \times \frac{\partial \mathbf{p}}{\partial \mathsf{u}_2} &= \sum_{\mathsf{i}\in\mathsf{I}} \Delta_{1,\mathsf{i}}(\mathbf{p}) b_\mathsf{i}(\mathsf{u}) \times \sum_{\mathsf{j}\in\mathsf{I}} \Delta_{2,\mathsf{j}}(\mathbf{p}) b_\mathsf{j}(\mathsf{u}) \\ &\in \operatorname{cone}(\Delta_{1,\mathsf{i}}(\mathbf{p}) \times \Delta_{2,\mathsf{j}}(\mathbf{p}))_{\mathsf{i}\in\mathsf{I},\mathsf{j}\in\mathsf{I}}.\end{aligned}$$

where u_i is the parameter in the direction e_i. □

It remains to characterize the relation between the mesh differences and normals near the extraordinary point of a subdivision surface. Including all directions radiating from the extraordinary point leads to poor bounds where normal directions and their negations are present. When the parameters are not near enough to the extraordinary point, where the characteristic map dominates, triangle normals can even reverse orientation under Loop subdivision (Figure 4, *left*). It is therefore not surprising that one can devise a configuration where the limit normal is the reverse of all initial facet normals (Figure 4, *right*). Such examples justify more intricate bound calculations such as pseudo-normal surfaces [6] or iteration-until-convergence based on eigendecomposition, subdivision and interval estimates as in [5].

Acknowledgements: This work was supported by NSF Grant CCF-0430891.

References

[1] I. Ginkel, J. Peters, G. Umlauf, On normals and control nets, in: The Mathematics of Surfaces IX, Loughborough, R. Martin and H. Bez and M. Sabin, pp. 233–239, 2005.

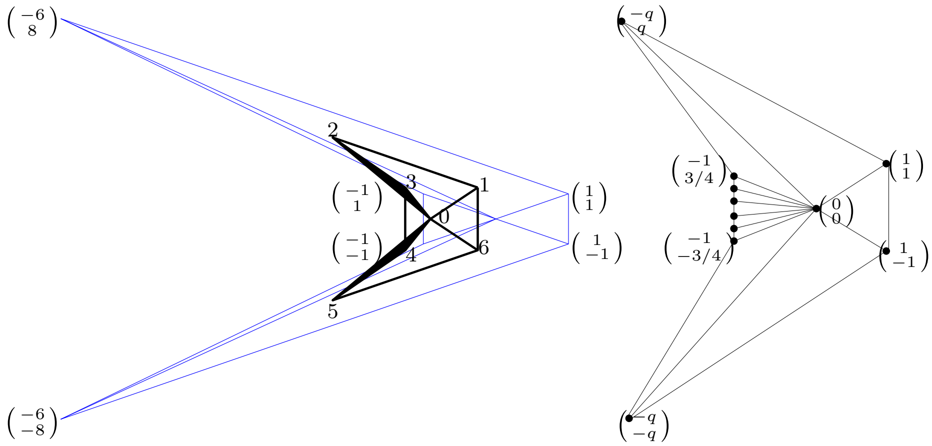

Fig. 4. Planar triangulations. (*left*) On input of the triangulation whose coordinates are given, the refined (*solid*) triangle with vertices numbered $2, 3, 0$ reverses orientation. (However, the limit normal at the central point is $(0, 0, 1)^T$, the same as all initial triangle normals.) (*right*) For this configuration (not to scale), the limit normal at the extraordinary point is $(0, 0, 1)^T$ if $q = 1$ and $(0, 0, -1)^T$ if $q = 10$.

[2] C. T. Loop, Smooth subdivision surfaces based on triangles, Master's thesis, Department of Mathematics, University of Utah (1987).

[3] M. Halstead, M. Kass, T. DeRose, Efficient, fair interpolation using Catmull-Clark surfaces, in: Computer Graphics Proceedings, 1993, pp. 35–44.

[4] C. de Boor, K. Höllig, S. Riemenschneider, Box splines, Springer, 1993.

[5] Eitan Grinspun, Peter Schröder, Normal Bounds for Subdivison-Surface Interference Detection, in: Proceedings Visualization 2001, Thomas Ertl and Ken Joy and Amitabh Varshney, eds., pp 333–340.

[6] Merlin Hughes, Christopher DiMattia, Ming C Lin, Dinesh Manocha, Efficient And Accurate Interference Detection For Polynomial Deformation, Proceedings Computer Animation, IEEE Computer Society, 1996, p155.