Smoothing polyhedra made easy

by

Jörg Peters†
jorg@cs.purdue.edu

Key words: C^1 surface, corner cutting, box splines, C^1 spline mesh, blending, vertex-degree, polyhedral

Running title: Smoothing polyhedra

Version: Sept 15 93 **Date printed:** August 3, 1995

to appear in: ACM TOG

Abstract

A mesh of points outlining a surface is polyhedral if all cells are either quadrilateral or planar. A mesh is vertex-degree bounded, if at most four cells meet at every vertex. This paper shows that if a mesh has both properties then simple averaging of its points yields the Bernstein-Bézier coefficients of a smooth, at most cubic surface that consists of twice as many three-sided polynomial pieces as there are interior edges in the mesh. Meshes with checker board structure, i.e. rectilinear meshes are a special case and result in a quadratic surface.

Since any mesh and, in particular any wireframe of a polyhedron can be refined, by averaging, to a vertex-degree bounded polyhedral mesh this result allows reinterpreting a number of algorithms that construct smooth surfaces and advertises the corresponding averaging formulas as a basis for a wider class of algorithms.

---

† Department of Computer Science, Purdue University, W-Lafayette IN 47907-1398
Supported by NSF grant CCR-9211322

1. Introduction

There are a variety of methods to give a smooth appearance to an object. For some animation purposes it suffices to hide the seams between components or to average the normals of the surface polyhedra during the shading process. For more sophisticated use, a good strategy is to adaptively refine the object by averaging the vertices of its boundary representation as in the subdivision algorithms [3] and [2]. The purpose of this paper is to show that with comparable effort one can obtain the explicit polynomial representation of a smooth, namely tangent continuous, surface that follows the outlines of a mesh of points such as the vertices and edges of a polyhedron. In particular, Section 2 below gives a simple recipe for smoothing vertex-degree bounded, polyhedral meshes whose definition is as follows.

- A mesh is polyhedral if all non quadrilateral mesh cells are planar.
- A mesh is vertex-degree bounded if at most four cells meet at every vertex.

Smoothing these special meshes is of interest, because any surface mesh can be turned into a vertex-degree bounded, polyhedral mesh by refinement and projection. A typical example of a refinement that results in a vertex-degree bounded mesh is the application of a single step of the Doo-Sabin algorithm [3]. Here a new mesh point is created for each vertex of each cell; then each new point is connected to the two new points corresponding to the neighboring vertices of the same cell and to the two new points corresponding to the neighboring cells and the same vertex. Consequently, each interior vertex has four neighbors at the end of the step. An example of a projection is the map $P_j \mapsto \sum P_i/n + 2\sum \cos(2\pi(i+j)/n)P_i/n$, for $j = 1..n$ (cf. [6]) which maps n points $P_1, \ldots, P_n$ into a common plane. Both operations can be scaled such that the basic shape of the mesh and hence of the resulting surface is preserved [13].

A number of recent algorithms for geometric design [7,8,11–14] are directly or indirectly based on reduction to a specific vertex-degree bounded, polyhedral meshes (see for example Figure 5). Once the mesh is obtained, the formulas of Section 2 apply. Besides reinterpreting existing algorithms, the formulas can serve as a guide to creating meshes that can be efficiently smoothed or locally modified to be efficiently smoothable. For example, chopping off a high-degree vertex that is convex with regard to its neighbors bounds the vertex-degree and creates locally a polyhedral mesh by a purely local operation. There are also meshes to which the smoothing algorithm can be applied directly. Buckminsterfullerene geodesic structures found in nature [17] are naturally vertex-degree bounded polyhedral due to the valence and energy distribution of carbon based structures.

This paper consists of the algorithm, its analysis and examples. The analysis is made easy by the fact that the planar cells serve as tangent planes at user-specified points. Should the mesh additionally be regular, then the C^1 surface becomes the quadratic defined by these tangent planes, a four-direction box spline surface [1]. For comparison, we note that the well-known algorithms [15] and [16] attack the more difficult problem of interpolating a mesh of curves. The polynomial degree of the resulting surfaces is therefore generally higher for these algorithms and the surfaces are more likely to oscillate [9, Section 2].

2. An algorithm for smoothing a vertex-degree bounded polyhedron

The input to the algorithm are points W_l and their neighbor relationships. The points W_l form a mesh consisting of quadrilateral and planar cells such that at most four cells meet at any interior mesh point. Since the construction averages the mesh points the planar cells should preferably be convex. Let e be the number of interior edges of the input mesh. The output of the algorithm are the Bernstein-Bézier coefficients $P_{ijk,l} \in \mathbb{R}^3$ of $2e$ three-sided cubic patches p_l:

$$p_l(u,v) := \sum_{i+j+k=3} P_{ijk,l} \frac{3!}{i!j!k!} u^i v^j w^k \qquad u,v \in [0,1], w := 1-u-v.$$

(cf. [4]) of a C^1 surface that follows the outline of the input mesh. By default, a global surface boundary consists of the boundaries of the well-defined patches one removed from the global mesh boundary. This is illustrated in Figure 1 and 3. Using this rule, we can match particular boundary data by adjusting or extending the mesh.

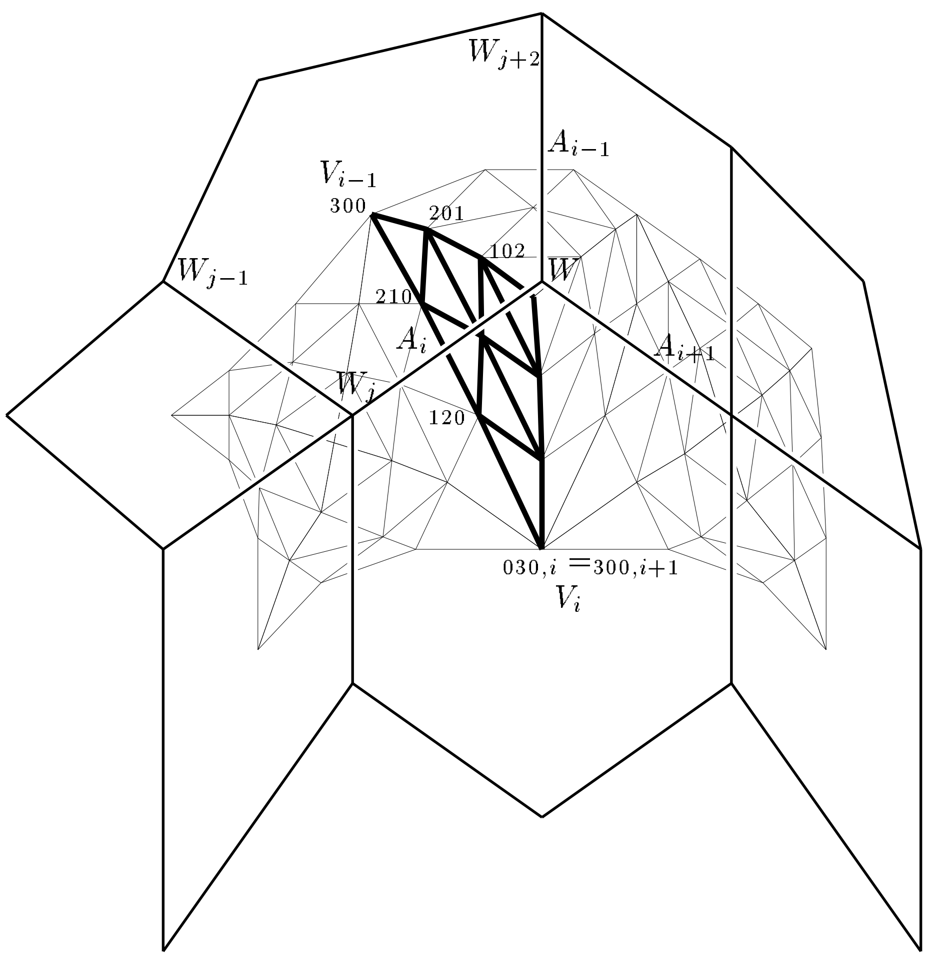

Figure 1. A mesh of points W_l and the resulting quilt of 3-sided patches.
The control net and the labels of the control points of one patch are emphasized.

The algorithm has two steps.

1. On each cell, choose a point V and on each edge choose a point A. For nonplanar (hence quadrilateral) cells with vertices W_j, $j = 1..m$, V should be the centroid and each A_j an edge midpoint, i.e.

$$V := \sum W_j/m \qquad A_j := (W_j + W_{j+1})/2, \qquad j = 1..m.$$

2. Each edge $i = 1..n$, $n \in \{3,4\}$ emanating from a mesh point W gives rise to a patch as follows. Let A_{i-1}, A_i and A_{i+1} be the points on consecutive edges, and V_{i-1}, V_i the points in the faces that share A_i. Then the ith patch corresponding to the vertex has the following coefficients.

$$P_{300,i} = V_{i-1} \qquad P_{210,i} = (2A_i + V_{i-1})/3 \qquad P_{120,i} = (2A_i + V_i)/3 \qquad P_{030,i} = V_i$$

if W has	$n = 3$ neighbors	$n = 4$ neighbors
$P_{201,i} =$	$(P_{300,i} + P_{210,i} + P_{120,i-1})/3$	$(P_{210,i} + P_{120,i-1})/2$
$P_{111,i} =$	$(A_{i-1} + A_{i+1} + 5A_i + V_{i-1} + V_i$ $+3\ell(V_{i-1} + V_i - 2A_i))/9$	$(A_{i-1} + A_{i+1} + 4A_i$ $+2\ell(V_{i-1} + V_i - 2A_i))/6$
$P_{021,i} =$	$(P_{030,i} + P_{120,i} + P_{210,i+1})/3$	$(P_{120,i} + P_{210,i+1})/2$
$P_{102,i} =$	$(P_{201,i} + P_{111,i} + P_{111,i-1})/3$	$(P_{111,i} + P_{111,i-1})/2$
$P_{012,i} =$	$(P_{021,i} + P_{111,i} + P_{111,i+1})/3$	$(P_{111,i} + P_{111,i+1})/2$

$$P_{003,i} = \frac{1}{n}\sum_{k=1}^{n} P_{012,k}$$

This completes the algorithm except for the specification of ℓ. Let Q be the patch abutting P labeled so that $Q_{ij0} = P_{ij0}$. Then $\ell := l_0 - l_1$, where l_0 and l_1 solve

$$\begin{aligned} l_0(P_{210} - P_{300}) &= m_0(P_{201} - P_{300}) + (1 - m_0)(Q_{201} - Q_{300}) \\ l_1(P_{030} - P_{120}) &= m_1(P_{021} - P_{120}) + (1 - m_1)(Q_{021} - Q_{120}). \end{aligned} \tag{$*$}$$

That is, l_i is the relative length of the projection of two transversal tangents onto the versal tangent. For example, if the cell associated with V_{i-1} is an affine n-gon and the cell associated with V_i is an affine m-gon, then $2\ell = \cos(2\pi/n) + \cos(2\pi/m)$.

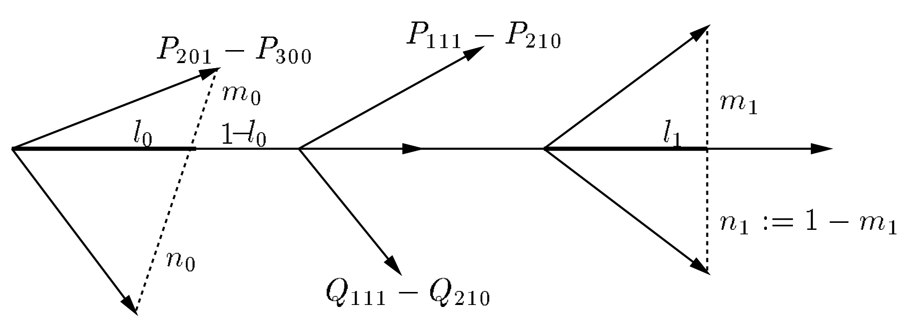

Fig. 2: Defintion of l_0 and l_1.

3. Properties of the smoothed surface

A number of properties of the smoothed surface are easy to establish. Since the vertex of each patch, $V = P_{300}$ is interpolated, the surface can interpolate one point on each planar cell. Since a cell is either planar or we choose its centroid and the edge midpoints, all A_i of a cell lie in the same plane, the tangent plane at V. Hence all planar cells are tangent planes of the surface. If all the coefficients $P_{ijk,l}$ are convex combinations of the mesh points then the surface lies in the *convex hull* of the polyhedral mesh. We see that the former is the case if $\ell \in [-1/3, 5/6]$ for $n = 3$ and $\ell \in [0,1]$ for $n = 4$. If more is known about the structure of the mesh, the convex hull property can be established independent of ℓ (see the Remark below). It remains to prove the tangent plane continuity of the resulting surface.

Theorem. *The piecewise cubic surface generated by the algorithm is C^1.*

Proof Using the abbreviation $[c_0, c_1, \dots, c_d]$ for the polynomial $\sum_{i=0}^{d} c_i(1-t)^{d-i}t^i$, we define the scalar polynomials a and b and the vector valued polynomials D_1p, D_2p, D_2q as follows.

$$
\begin{aligned}
n_0 &:= 1 - m_0, \quad n_1 := 1 - m_1, \\
a &:= [1,1], \quad b := [\,\frac{l_0}{n_0}, 2(\frac{l_0}{n_1} + \frac{l_1}{n_0}), \frac{l_1}{n_1}\,], \quad c := -[\,\frac{m_0}{n_0}, \frac{m_1}{n_1}\,] \\
D_1p &:= 2[A_i - V_{i-1}, V_i - A_i] \\
D_2p &:= 3[P_{201,i} - P_{300,i}, 2(P_{111,i} - P_{210,i}), P_{021,i} - P_{120,i}] \\
D_2q &:= 3[Q_{201,i} - Q_{300,i}, 2(Q_{111,i} - Q_{210,i}), Q_{021,i} - Q_{120,i}]
\end{aligned}
$$

Here D_1p is short for the versal derivative $\frac{\partial p}{\partial t}(t,0)$ and expands to $2(A_i - V_{i-1})(1-t) + 2(V_i - A_i)t$. Similarly, D_2p and D_2q represent the transversal derivatives of the patches along the common boundary. It suffices to show that all three derivatives lie in a common plane, i.e. that

$$aD_2q = bD_1p + cD_2p.$$

A cusping match of the patches is ruled out because $a(t)c(t) < 0$ (cf. [10, Lemma 2.1]). Equating the four coefficients of the cubic polynomials on either side, we find that the first and the last coefficients agree due to Equation $(*)$ while the remaining two must be verified by lengthy but standard substitution of A_l and V_l, using $k_P := \begin{cases} 1/3 & \text{if } n = 3 \\ 1/2 & \text{if } n = 4 \end{cases}$ and

$$
\begin{aligned}
P_{201,i} - P_{300,i} &= 2k_P(A_{i-1} + A_i - 2V_{i-1})/3 \\
P_{021,i} - P_{120,i} &= 2((1-2k_P)V_i + k_P A_{i+1} + (k_P - 1)A_i)/3.
\end{aligned}
$$

■

Remark (for the expert). The two equations associated with the second and third coefficient of $bD_1p + cD_2p - aD_2q$ in the unknowns P_{111} and Q_{111} have a family of solutions parametrized by r (the formula for Q_{111} follows by symmetry):

$$\begin{aligned}3P_{111} :=& \frac{k_P}{r}A_{i-1} + k_P r A_{i+1} + ((l_1 - 1 + k_P)r^2 + (2-\ell)r + (k_P - l_0))/rA_i \\ &+ ((1-l_1)r + l_0 - 2k_P)/rV_{i-1} + ((1 - l_1 - 2k_P)r + l_0)V_i.\end{aligned}$$

The algorithm uses $r = 1$ to maximize symmetry of the formulas. If more is known about the mesh, an asymmetric choice can be preferable. For example, in [13], the cells containing V_{i-1} and V_i are known to be a quadrilateral, i.e. $n = 4$, and an affine m-gon. Therefore $l_0 = 0$ and $\ell = -l_1 = \cos(2\pi/m)/2$, and the parameter r is chosen so that the convex hull property holds also for $m = 3$.

Corollary. *If $n = 4$ and $\ell = 0$, then the patch generated by the algorithm is quadratic.*

Proof Since the boundary curve is a degree-raised quadratic and

$$6P_{111} := A_{i-1} + A_{i+1} + 4A_i + 2\ell(V_{i-1} + V_i - 2A_i),$$

the construction yields a degree-raised quadratic polynomial patch with coefficients

$$\begin{array}{ccccc} & & P_{002} & & \\ & P_{101} & & P_{011} & \\ P_{200} & & P_{110} & & P_{020} \end{array} = \begin{array}{ccccc} & & \frac{1}{4}\sum A_j & & \\ & \frac{A_{i-1}+A_i}{2} & & \frac{A_{i+1}+A_i}{2} & \\ V_{i-1} & & A_i & & V_i \end{array}$$

■

To check the formulas, I implemented the algorithm and generated surfaces from meshes. Figure 2 shows a surface resulting from a vertex-degree-bounded polyhedral mesh that has a boundary. Figure 3 shows a closed surface, a Christmas ornament if you like. Figure 4 illustrates the conversion of an original mesh to a special type of vertex-degree bounded polyhedral mesh. Here the refinement has been chosen intentionally to round all features and create a "soft" vice. A more functional smoothing of the vice data, using locally adjusted change of curvature, is shown in [13].

References

[1] C. W. de Boor, K. Höllig, S. Riemenschneider, *Box splines*, Springer Verlag, NY, 1994.

[2] E. Catmull, J. Clark, Recursively generated B-spline surfaces on arbitrary topological meshes, *Computer Aided Design* **10**, No 6, 1978: 350–355.

[3] D. Doo, A subdivision algorithm for smoothing down irregularly shaped polyhedrons, *Proceedings on interactive techniques in computer aided design*, Bologna, 1978: 157–165.

[4] G. Farin, Curves and surfaces for computer aided geometric design, Academic Press, 1990.

[5] T.N.T. Goodman, Closed surfaces defined from biquadratic splines, Constructive Approximation 7 1991, 149-160.

[6] C. T. Loop, Smooth subdivision surfaces based on triangles, Master's thesis, University of Utah, 1987.

[7] S.L. Lee, A. A. Majid, Closed smooth piecewise bicubic surfaces, ACM Transactions on Graphics 10,4 (1991): 342–365.

[8] C. T. Loop, Smooth spline surfaces over irregular meshes, Siggraph 1994.

[9] S. Mann, Surface approximation using geometric Hermite patches, thesis, University of Washington, Dept. of Computer Sciences, 1992.

[10] J. Peters, Smooth Interpolation of a Mesh of Curves, Constructive Approximation, 7 (1991), 221-247.

[11] J. Peters, Constructing C^1 surfaces of arbitrary topology using biquadratic and bicubic splines, May 1992, in *Designing fair curves and surfaces*, N. Sapidis (ed.), to appear.

[12] J. Peters, Smooth free-form surfaces over irregular meshes generalizing quadratic splines, Sept 1993, CAGD **10**, 1993:347–361.

[13] J. Peters, C^1-surface splines, *SIAM J. of Num. Anal.* **32**,2: 645–666.

[14] A.R.M. Piah, Construction of smooth surfaces by piecewise tensor product polynomials, *Report CS 91-04, University of Dundee.* Feb 1991.

[15] B. Piper, Visually smooth interpolation with triangular Bézier patches, in: Geometric Modelling: Algorithms and trends, G.E. Farin (ed.) 221–233. SIAM, 1987.

[16] L. Shirman, C. Séquin, Local surface interpolation with Bézier patches, Computer Aided Geometric Design 4 : 279–295. Errata Computer Aided Geometric Design 6.

[17] R. E. Smalley, The third form of Carbon, *Naval Research Reviews*, **3** 1991.