Eurographics Symposium on Geometry Processing (2004)
R. Scopigno, D. Zorin, (Editors)

Differentiable Parameterization of Catmull-Clark Subdivision Surfaces

Ioana Boier-Martin[1] and Denis Zorin[2]

[1] IBM T. J. Watson Research Center, [2] New York University

Abstract

Subdivision-based representations are recognized as important tools for the generation of high-quality surfaces for Computer Graphics. In this paper we describe two parameterizations of Catmull-Clark subdivision surfaces that allow a variety of algorithms designed for other types of parametric surfaces (i.e., B-splines) to be directly applied to subdivision surfaces. In contrast with the natural parameterization of subdivision surfaces characterized by diverging first order derivatives around extraordinary vertices of valence higher than four, the derivatives associated with our proposed methods are defined everywhere on the surface. This is especially important for Computer-Aided Design (CAD) applications that seek to address the limitations of NURBS-based representations through the more flexible subdivision framework.

1. Introduction

A number of properties make subdivision surfaces attractive for graphics applications, including their ability to handle general control mesh topologies, the relatively high visual quality of the resulting surfaces for a broad range of input meshes, and the efficiency of their evaluation. These same properties make subdivision surfaces an appealing choice for traditional CAD systems, primarily as a tool for conceptual design. However, as observed in [GN01], a number of problems must be resolved in order to integrate a new surface type into an existing CAD system.

Commercial CAD systems tend to have a complex multilayered architecture and must support a variety of legacy features and representations. For example, most systems employ B-splines as the main parametric surface representation. Many of the algorithms they implement (e.g., tessellation and surface-surface intersection [BFJP87, BHLH88, Hoh91, KPW92]) assume, either implicitly or explicitly, properties specific to splines. In many instances, however, the actual surface representation is hidden by a software abstraction layer and could be replaced with a different representation, provided that standard parametric evaluators are supplied as an interface with the rest of the system.

In this paper we describe two parameterizations of Catmull-Clark subdivision surfaces [CC78] which makes them compatible with a typical boundary representation abstraction. Specifically, the topology of the surface is defined by a mesh consisting of quadrilateral faces; each face has a surface patch associated with it; each surface patch is parameterized over a standard parametric domain $[0,1]^2$ with coordinates (u,v) and derivatives are defined everywhere on D. The first parameterization is efficient and easy to implement, however, the derivatives vanish at the corners of the domain. The second parameterization is more complex aa it requires the inversion of the characteristic map, however, it guarantees a non-degenerate Jacobian.

The parameterizations described in this paper have been tested in the context of the CATIA solid modeler [†].

1.1. Previous Work

An algorithm for explicit evaluation of subdivision surfaces at arbitrary points was first proposed in [Sta98] for Catmull-Clark surfaces and in [Sta99] for Loop subdivision surfaces [Loo87]. This development was a part of the effort to integrate subdivision surfaces into Alias|Waverfront Maya. An extension of this method is presented in [ZK02]. Direct evaluation is useful in many applycations (e.g., see

[†] http://www.catia.ibm.com

[GS01, KP03]). Our work builds on Stam's approach. A major shortcoming of the parameterization used by Stam is that it has has unbounded derivatives around extraordinary vertices, resulting in the failure of algorithms which assume nonsingular surface parameterizations.

Several papers discuss algorithms for various modeling operations directly applicable to subdivision surfaces (e.g. [KS99, LLS01, BLZ00]). The goal of this work is different: we describe a well-behaved parameterization of subdivision surfaces which makes it possible to take advantage of *existing* algorithms that were *not* originally intended for subdivision surfaces. We note that special-purpose algorithms are likely to perform better and take better advantage of the important features of subdivision surfaces. However, this approach is not always practical in the context of existing systems.

1.2. Overview

The starting point of our construction is the parameterization used in [Sta98], which we call *natural*. It is described in Section 2. The main idea is to compose the natural parameterization with a one-to-one reparameterization map from the unit square D to itself to ensure the desired properties of the derivatives. If a face has a single extraordinary vertex, this reparameterization is constructed as the inverse of an approximation of the characteristic map, or the inverse of the characteristic map itself, blended with the identity map. We obtain the desired boundedness of the derivatives by construction. The main theoretical challenge in this work (see Appendix A) is to establish that the resulting blended map of the unit square onto itself is one-to-one.

Notation. We use $f(u,v)$ to denote the natural parameterization of a patch of a subdivision surface over the square $[0,1]^2$. We use $y = (r,s)$ to refer to points in the domain of the natural parameterization. The points in the domain of the reparameterization map from the unit square to itself are denoted by $t = (u,v)$.

2. The Natural Parameterization

The goal of this section is to introduce the basic notations used for subdivision surfaces and to define the natural parameterization that can be evaluated using Stam's algorithm [Sta98].

Control meshes. A subdivision surface can be viewed as a function defined on the faces of the control mesh. In this paper, we restrict our attention to control meshes with quadrilateral faces. Each face can be identified with a square domain $[0,1]^2$ with coordinates (r,s).

More precisely, we consider subdivision surfaces associated with control meshes $(K,\mathbf{p})$, where $K = (V,E,F)$ is a two-dimensional polygonal complex, and $\mathbf{p}$ is the vector of values assigned to the vertices of K (typically the values belong to $\mathbf{R}^n$ for $n = 1,2,3$). The set of faces F is a set of quadruples of vertices and the edges are unordered pairs of vertices. If (v_1,v_2,v_3,v_4) is a face, (v_i,v_{i+1}) are its edges, for $i = 1\dots 4$ (the index increment is defined modulo 4). The set of edges E is the union of the sets of edges of all faces.

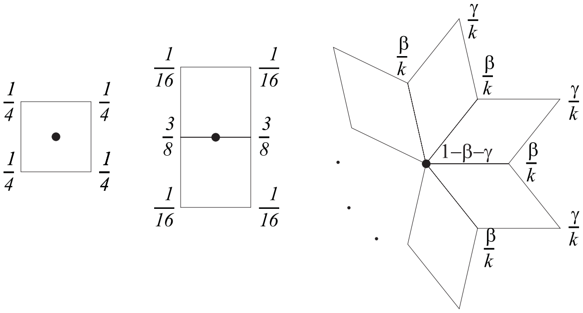


Figure 1: *Catmull-Clark subdivision rules. Left: the rule for a vertex added at the centroid of a face. Middle: the rule for vertex added at an edge. Right: the rule for the update of an existing vertex,* $\beta = 3/2k$, $\gamma = 1/4k$, *k is the vertex valence.*

A 1-neighborhood $N_1(v)$ of a vertex v in K is the subcomplex formed by all faces that have v as a vertex. The m-neighborhood of a vertex v is defined recursively as a union of all 1-neighborhoods of vertices in the $(m-1)$-neighborhood of v.

Recall that a *link* of a vertex is the set of edges of $N_1(v,K)$ that do not contain v. We consider only complexes corresponding to manifold meshes, i.e., with all vertices having links that are simple connected polygonal lines, either open or closed. If the link of a vertex is an open polygonal line, the vertex is a *boundary vertex*, otherwise it is an *interior vertex*.

Subdivision. We can construct a new complex $D(K)$ from a complex K through subdivision by splitting each quadrilateral face in four. A new vertex is added for each old face and edge, and each face is replaced by four new faces. We use K^j to denote the j times subdivided complex $D^j(K)$ and V^j to denote the set of vertices of K^j. A subdivision scheme defines for each complex K a linear mapping from the values $\mathbf{p}$ on V to new values $\mathbf{p}^1$ defined on V^1. For most common subdivision schemes, for any vertex $v \in V$ this mapping (i.e., the subdivision rule) depends only on the structure of the neighborhood of v. In this paper, we consider Catmull-Clark surfaces with rules as shown in Figure 1.

Limit functions and natural parameterization. If all faces of the original control mesh are planar and the mesh has no self intersections, the subdivision surface can be regarded as a function on the mesh which we denote by $|K|$. However, these assumptions are somewhat restrictive, so it is best to construct the domain $|K|$ in a more abstract way. We consider a disjoint union of copies of the unit square $[0,1]^2$, one for each face. For each pair of faces sharing an edge, we identify the edges of the corresponding squares. We denote the domain obtained in this way by $|K|$ and we refer to the corresponding squares as *faces* of $|K|$. Subdivision surfaces can be defined as functions on $|K|$:

Definition 1 For any given control point vector $\mathbf{p}$, bilinear interpolation defines a bilinear function on each face of $|K|$, which we denote $f[\mathbf{p}]$. *The subdivision limit function* is the pointwise limit of the sequence of functions $f[\mathbf{p}^i]$ defined in $|K|$, whenever this sequence converges. We call the limit function the *natural parameterization* of the limit surface. For the Catmull-Clark rules limit functions are defined for any control mesh.

The image of the limit function is usually called the *limit subdivision surface.* Sometimes, the same term is used to denote the limit function itself. As the distinction is important in the context of our paper, we always differentiate between the limit surface (a subset of $\mathbf{R}^3$) and the limit function (a map from $|K|$ to $\mathbf{R}^3$).

Our goal is to improve the natural parameterization. As shown in [Sta98] it can be evaluated in constant time at any point in $|K|$. As we demonstrate next, the main problem with this parameterization is that its derivatives typically diverge near extraordinary vertices. We address this issue by reparameterization, i.e., by considering compositions $f[p] \circ T$ where T is a one-to-one map from $|K|$ onto itself. Furthermore we consider only reparameterizations which preserve faces of $|K|$. In this case T restricted to each face is a one-to-one map of the unit square onto itself (Section 4).

3. Derivative Behavior Near Extraordinary Vertices

We briefly review the theoretical tools necessary to examine the behavior of the derivatives of the natural parameterization at extraordinary vertices: subdivision matrices and eigenbasis functions. We make use of these tools to define our reparameterizations.

Due to the locality of subdivision rules, local properties of subdivision surfaces can typically be examined on k-regular complexes instead of general meshes. A complex is called *k-regular* if it has a single extraordinary vertex (i.e., a vertex of valence $\neq 4$). We denote such complex $\mathcal{R}_k$.

Subdivision matrices. The key to the analysis of stationary subdivision is the *subdivision matrix*. Eigenvalues and eigenvectors of this matrix are closely related to the smoothness properties of subdivision. Consider the part of a subdivision surface $f[p](y)$ with $y \in U_1$, where U_1 is the k-star formed by the faces of the k-regular complex $\mathcal{R}_k$ adjacent to the central vertex. It is straightforward to show that the limit surface on this k-star can be computed given the control points $p^0(v)$ for all vertices v in 2-neighbrhood N_2 of the extraordinary vertex c. In particular, the control points $p^1(v)$ for vertices in the 2-neighborhood of c in the *subdivided* complex $\mathcal{R}^1$ can be computed using only control points $p^0(v)$. Same is true if we consider subdivision levels j and $j+1$ instead of 0 and 1.

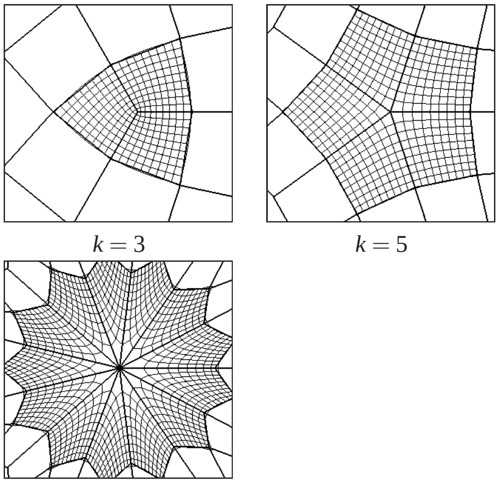


Figure 2: *Behavior of the isoparameter lines associated with the natural parameterization near extraordinary vertices.*

Definition 2 Let $\mathbf{p}^j$ be the vector of control points $p^j(v)$ for $v \in N_2(0, \mathcal{R}_k^j)$. Let $Q+1 = 6k+1$ be the number of vertices in $N_2(0, \mathcal{R}_k)$. As the subdivision rules are linear and do not depend on the level j, $\mathbf{p}^{j+1}$ can be computed from $\mathbf{p}^j$ using a $(Q+1)\times(Q+1)$ matrix S: $\mathbf{p}^{j+1} = S\mathbf{p}^j$. S is called the Catmull-Clark subdivision matrix.

Eigenvalues and eigenvectors of this matrix have fundamental importance for smoothness of subdivision.

Eigenbasis functions. Let $\lambda_0 = 1, \lambda_1, \ldots \lambda_{Q+1}$, be eigenvalues of the subdivision matrix in nonincreasing order of magnitude each repeated according its multiplicity (we assume that the matrix is nonderogatory, i.e. the total number of eigenvalues counted with multiplicities is $Q+1$). For Catmull-Clark and other commonly used subdivision schemes, $1 = \lambda_0 > \lambda_1 = \lambda_2 > \lambda_3$. Furthermore, all eigenvalues are real and the eigenvector b^i corresponding to λ_i satisfies $Sb^i = \lambda_i b^i$.

Definition 3 The *eigenbasis functions* are the limit functions of subdivision on the k-star defined by $f^i = f[b^i] : U_1 \to \mathbf{R}$.

A Catmull-Clark subdivision surface $f[\mathbf{p}](y)$ defined on U_1 can be represented as

$$f[\mathbf{p}](y) = \sum_i \beta^i f^i(y) \qquad (1)$$

where $\beta^i \in \mathbf{R}^3$. One can show using the definition of limit functions of subdivision that the eigenbasis functions satisfy the following *scaling relations*:

$$f^i(y/2) = \lambda_i f^i(y) \tag{2}$$

It is often convenient to regard the eigenbasis functions restricted to a single quad in the k-star using the bilinear coordinates of the quad. The scaling relation (2) still holds in this case and we use notation $f^i(r,s)$ for this linear reparameterization of the eigenbasis functions.

Divergence of derivatives. Equations (1) and (2) allow us to demonstrate immediately that the derivatives of $f[p](y)$ diverge for almost any $\mathbf{p}$. Indeed, differentiating with respect to u, we obtain:

$$\frac{\partial f^i}{\partial r}(r/2, s/2) = 2\lambda_i \frac{\partial f^i}{\partial r}(r,s). \tag{3}$$

For the Catmull-Clark scheme the scaling factor $s_f = 2\lambda_1$ is greater than one for valences greater than four. If we choose a point (r_0, s_0) for which $\frac{\partial f^1}{\partial r}(r,s)$ is nonzero (existence of such points follows from the analysis of the characteristic map for the Catmull-Clark scheme in [PR98]), then the value of the derivative for points $(r_0/2^m, s_0/2^m)$ converging to $(0,0)$ as $m \to \infty$ increases at the rate $(2\lambda_i)^m$. A similar proof can be used for the derivative with respect to v. We conclude that for any surface for which β^1 is nonzero, the derivatives of the natural parameterization diverge.

4. Differentiable Parameterizations

As Catmull-Clark subdivision surfaces are C^1-continuous at the extraordinary vertices ([PR98]), a C^1 parameterization exists for any surface patch corresponding to a face. Our goal is to construct this parameterization explicitly. We base our construction on the blending of parameterizations.

4.1. Blending Parameterizations

The unit square can be regarded as the intersection of four quadrants. We are going to construct new parameterizations for subdivision surfaces using the natural parameterization over the unit square composed with a map of the unit square to itself. These maps will be constructed by blending *quadrant maps*, i.e., maps defined on each quadrant.

Specifically, let $F_i(t)$ $i = 1,2,3,4$, be maps from the quadrant $\mathbf{R}^2_+ = \{t = (u,v) | u \geq 0, v \geq 0\}$, to itself. We use these maps to "fix" the divergence of the derivatives: we choose them such that $f \circ F_i$ has derivatives everywhere including at zero ($f(r,s)$ is a limit function of subdivision restricted to a single face). However, the F_i's do not necessarily map the unit square to itself. We use a blend of such maps to generate a new map that has this property and therefore yields the desired face reparameterization.

Let v_i be the vertices of a the unit square $[0,1]^2$. Let T_i be the orthogonal transformations of the plane mapping the interval $[0,1]$ on two coordinate axes to the sides of the unit square adjacent to vertex v_i. Let $w : [0,1]^2 \to \mathbf{R}$ be a weight function which satisfies the boundary conditions $w(0,0) = 1$, $w(u, 1/2) = w(1/2, v) = 0$, $w' = w'' = 0$ for the boundary of $[0,1/2]^2$, and $w(u,v) \equiv 0$ outside $[0,1/2]^2$. Define $w_i = w \circ T_i^{-1}$, i.e. the weight function rotated so that it is 1 at v_i. We define the blend of the maps F_i and the identity as follows:

$$F_b = \sum_i w_i \, T_i \circ F_i \circ T_i^{-1} + (1 - w_i) Id. \tag{4}$$

The idea of this definition is straightforward: we gradually decrease the influence of the map F_i as we move away from a vertex v_i; outside the quarter of the square adjacent to v_i it has no influence. Away from the corners, where there is no need to alter the derivatives of the parameterization, the reparameterization is close to the identity map.

Our choice of conditions on the blending functions ensures that the properties of the derivatives of F_i at corners are preserved. However the fact that F_i are one-to-one does not guarantee that their blend is also one-to-one. We reduce the problem of verifying that F_b is one-to-one to checking the sign of the Jacobian using the following proposition.

Proposition 1 If the Jacobian of the map F_b does not vanish away from the corners and if F_i are one-to-one maps, then F_b is one-to-one map from $[0,1]^2$ onto itself.

The proof of this Proposition is given in Appendix A.

4.2. Parameterization with Vanishing Derivatives

Our first reparameterization uses simple closed-form functions F_i of the following form:

$$F^\alpha(t) = |t|^{\alpha - 1} t,$$

We are interested in values of α for which the scaling factor s_f of equation (3) after reparameterization is strictly less than one. This ensures that the values of the derivatives for points converging to the extraordinary vertex will converge to zero after reparameterization.

To derive a valid range for α we use polar coordinates. Let (ρ, φ) denote the polar coordinates of $t = (u,v)$. The map F becomes $F^\alpha(\rho, \varphi) = (\rho^\alpha, \varphi)$ and simple substitution in equation (3) yields the following scaling relation after reparameterization:

$$\frac{\partial h^i}{\partial u}(\rho/2^{1/\alpha},\varphi) = 2^{1/\alpha}\lambda_i \frac{\partial h^i}{\partial u}(\rho,\varphi), \qquad (5)$$

where $h^i(\rho,\varphi) = (f^i \circ F^\alpha)(\rho,\varphi)$. By imposing the constraint $2^{1/\alpha}\lambda_i < 1$ we infer that, if α is greater than $-\log 2/\log\lambda_i$, the values of the derivative $\frac{\partial h^i}{\partial u}$ converge to zero around the origin.

It is also straightforward to show that the blended map is one-to-one by using radially symmetric weight functions $w(t)$ defined as $b(|t|)$, where $b(u) = B(cu+1) + B(cu) + B(cu-1)$, $B(u)$ is the uniform cubic B-spline basis function, and $c = 1/6$.

Indeed, it is sufficient to consider each quarter of $[0,1]^2$ separately. The expression for the restriction of F_b is $(r,\varphi) \to (b(\rho)\rho^\alpha + (1-b(\rho))\rho, \varphi)$. To prove that it is one-to-one it is sufficient to show that $b(\rho)\rho^\alpha + (1-b(\rho))\rho$. Direct differentiation shows that the derivatives are positive and therefore the function is monotonically increasing.

Figure 4 illustrates the behavior of the iso-parameter lines around vertices of valence 5 and 13, respectively, after reparameterization with vanishing derivatives.

The advantage of this reparameterization is its simplicity. However, it is easy to see that it forces derivatives to vanish at the extraordinary vertices, which may be undesirable for some algorithms relying on the fact that parameterizations are non-degenerate, e.g., for tracing intersection curves. A more complex approach is needed in this case.

4.3. Parameterization with a Non-degenerate Jacobian

To obtain surface parameterizations with non-zero Jacobian we use *characteristic maps.* Characteristic maps were introduced in [Rei95] as a tool for the analysis of subdivision.

Definition 4 For a given valence k and a subdivision scheme with eigenvalues of the subdivision matrix for that valence satisfying $1 = \lambda_0 > \lambda_1 = \lambda_2 > \lambda_3$ and eigenfunctions f_1 and f_2 which do not vanish identically, the **characteristic map** $\Phi : U_1 \to \mathbf{R}^2$ is the map defined as (f^1, f^2).

In [Rei95] it was shown that, under the assumptions of the definition, if the characteristic map is regular (i.e., it has a non-vanishing Jacobian wherever the Jacobian is defined) and one-to-one, the subdivision scheme produces C^1-continuous surfaces for almost any choice of control points. As each component of the characteristic map is an eigenfunction, it satisfies the scaling relationship $\Phi(y/2) = \lambda\Phi(y)$ where $\lambda = \lambda_1 = \lambda_2$.

The characteristic map for symmetric schemes, such as Catmull-Clark, has the property of mapping the line segments from $(0,0)$ to $(cos(2j\pi/k), sin(2j\pi/k))$ to themselves for a suitable choice of eigenvectors b_1, b_2 corresponding to $\lambda_1 = \lambda_2 = \lambda$.

Similarly to eigenbasis functions, we can restrict Φ to a single quad and reparameterize it using bilinear coordinates (u,v) over the quad. Using the scaling relation, we can extend $\Phi(u,v)$ to $\mathbf{R}^2_+$: for any (u,v) let m be the smallest number m such that $(u/2^m, v/2^m)$ is inside U_1. Then $\Phi(u,v) = \lambda^{-m}\Phi(u/2^m, v/2^m)$. For the choice of b_1 and b_2 we have assumed, the extended $\Phi(u,v)$ is a map from $\mathbf{R}^2_+$ onto itself and can be used as a quadrant map F_i in (4).

Figure 3 illustrates the structure of layers and patches of a segment of a characteristic map. The map is polynomial on each patch, and the layers are similar under scaling.

We now consider quadrant maps of the form $F_j(t) = \Phi_j^{-1}(t)$, where Φ_j is the characteristic map for the valence of vertex v_j. As for almost any subdivision limit function, $f \circ \Phi_j^{-1}(t)$ is known to be differentiable at the extraordinary vertex, hence this approach ensures the desired property of the parameterization.

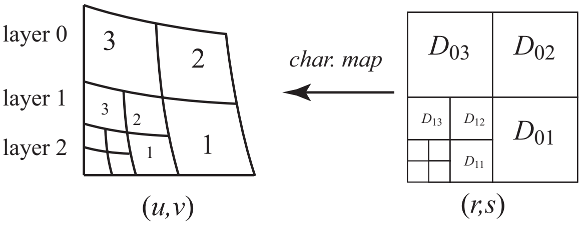


Figure 3: *The structure of layers and patches around the extraordinary vertex. The domains of the polynomial patches of the natural parameterization are shown on the right, whereas their characteristic map images appear on the left.*

To obtain the complete reparameterization map using (4), we define $w(t) = b(u)b(v)$ for $t = (u,v)$. The reason for using a different weight function is that it is easier to verify that the resulting map is one-to-one (see Appendix A). The idea of the proof is straightforward, however, it requires algebraic calculations using a computer algebra system.

The main practical challenge is the efficient computation of the inverse of the characteristic map. We take advantage of the structure of layers and patches induced by the scaling relation to directly evaluate $\Phi^{-1}(t)$. Specifically, it is well known (e.g., [Rei95]) that a subdivision surface based on splines (Catmull-Clark in particular) can be separated into an infinite number of polynomial patches. For the natural parameterization over a single face, these patches can be organized into L-shaped layers as shown in Figure 3. Each layer can be obtained from the previous one by scaling by λ. This means that it is possible to reduce the problem of inverting the characteristic map to inverting three polynomial patches P_1, P_2, P_3 defined on D_{01}, D_{02} and D_{03}. Let D_{l1}, D_{l2}, D_{l3}

be three domains of patches in a given layer l. The overall structure of the algorithm is as follows.

Algorithm 1:

Input: $t = (u, v)$; Output: $(r, s) = \Phi^{-1}(u, v)$

1. Determine layer number l of the characteristic map and polynomial patch number j, $j = 1, 2, 3$ in layer l such that $(u, v) \in \lambda^l P_j(2^l D_{lj})$
2. Compute $(r, s) = \Phi^{-1}(u, v) = 2^{-l} P_j^{-1}(\lambda^{-l} t)$

Layer and patch identification. As illustrated in Figure 3, the layers and patches in the image of the characteristic map are bounded by curves. Thus the first step of the algorithm (finding the layer and patch number for a point in the image), needs to test on which side of a curved boundary a point is. As each polynomial patch is bicubic, the boundaries are parametric cubic curves. For efficient inside-outside tests, we convert the curves to implicit form $g(u, v) = 0$. The sign of $g(u, v)$ determines on which side of the curve a given point (u, v) is.

While several methods exist for implicitization [SC95] we only need to implicitize cubic curves, and we opt for a simple approach based on *polynomial resultants* [Apo70].

Definition 5 Let $q(t) = q_m x^m + q_{m-1} x^{m-1} + \cdots + q_0$ and $r(t) = r_n x^n + r_{n-1} x^{n-1} + \cdots + r_0$ denote two scalar polynomials such that at least one of q_m and r_n is non-zero. The *Sylvester resultant* of the two polynomials is defined as the determinant of the following $(m+n) \times (m+n)$ *Sylvester matrix*:

$$S(q, r) = \begin{bmatrix} q_0 & q_1 & \cdots & q_{m-1} & q_m & 0 & 0 & \cdots & 0 \\ 0 & q_0 & \cdots & q_{m-2} & q_{m-1} & q_m & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & q_0 & q_1 & q_2 & q_3 & \cdots & q_m \\ r_0 & r_1 & \cdots & r_{n-1} & r_n & 0 & 0 & \cdots & 0 \\ 0 & r_0 & \cdots & r_{n-2} & r_{n-1} & r_n & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & q_0 & r_1 & r_2 & r_3 & \cdots & r_n \end{bmatrix}$$

A connection between the Sylvester resultant and the common roots the two polynomials is established by the following well-known result [GKZ96]: *Two polynomials q and r have a common root if and only if* $\det S(q, r) = 0$. To decide the position of a given point (u_0, v_0) with respect to a parametric polynomial curve $\mathcal{C}(t) = (u(t), v(t))$ we set $q(t) = u(t) - u_0$ and $r(t) = v(t) - v_0$ and we compare $\det S(q, r)$ to zero (or a small ε to account for numerical precision errors). To make the calculation more efficient, for an input point (u, v) we first compute an approximate layer number l' can be quickly computed as $l' = \lceil \log[\lambda] max(u, v) \rceil - 1$. We then use implicitization to determine a nearby patch which contains (u, v).

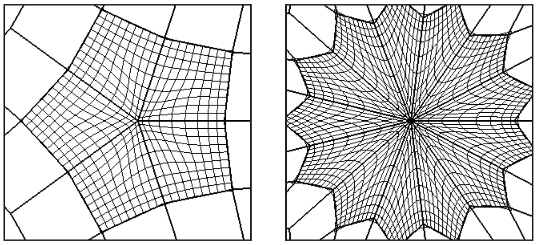

$k = 5$ $k = 13$

Figure 4: *Isoparameter lines near extraordinary vertices after reparameterization with vanishing derivatives.*

Patch inversion. After locating the layer and patch containing a given (u, v) we must compute its pre-image $(r, s) = \Phi^{-1}(u, v)$, which, as we have shown, only requires inversion of three bicubic patches P_1, P_2 and P_3. To compute the inverse of a bicubic patch we use Newton's method [Cd80] to find a root of the following equation inside our domain (i.e, the unit square):

$$P_j(r, s) - \begin{bmatrix} u \\ v \end{bmatrix} = 0 \qquad (6)$$

We use (u, v) as the initial approximation. Although the Newton method is not guaranteed to converge, we have observed consistent convergence, as our reparameterization introduces relatively little distortion. A more reliable approach would be to use Bezier clipping.

Figure 5 illustrates the behavior of the iso-parameter lines around extraordinary vertices after reparameterization by inversion of the characteristic map. Note the equal spacing between consecutive lines in contrast to Figures 2 and 4.

Figure 6 illustrates the application of our evaluation with reparameterization around extraordinary vertices on multiresolution subdivision surfaces corresponding to scanned objects. Figure 7 shows comparatively the magnitude of the derivatives around extraordinary vertices of various valences with and without reparameterization.

Computationally, the reparametrization introduces a significant overhead. The goal of our current implementation was correctness, rather than efficiency, and we can only estimate the relative cost of different stages. We have observed that the computation time is dominated by step 1 in Algorithm 1 (i.e. finding the layer and the patch). Step 2 (inversion of the polynomial patch) is much less expensive and takes a small fraction of the time it takes to evaluate a point without reparametrization. Our current implementation of step 1 is highly inefficient; with inefficiencies eliminated, based on the floating point operations counts, we expect that its cost will be lower than the cost of evaluation. However, we should caution the reader that the wall-clock timings depend strongly on the datastructures used for storing control

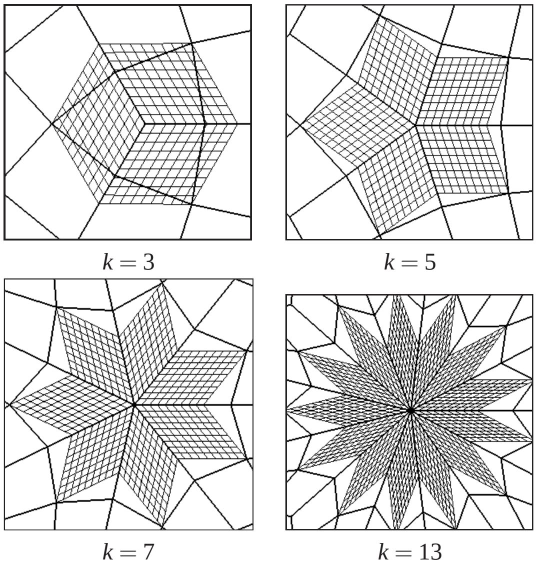


Figure 5: *Behavior of isoparameter lines near extraordinary vertices after reparameterization with a non-degenerate Jacobian.*

points, eigenvectors, and other necessary information and can vary in wide range.

Furthermore, we have considered random point evaluation, for which no coherence between sequential evaluations is assumed. In practice, it is more likely that many close points are evaluated, or that the surface is evaluated at the same points many times. In such cases many computed values can be cached. As a result the actual computational overhead of reparametrization can vary greatly, from negligible to high.

5. Conclusions and Future Work

In this paper we have described two methods for generating everywhere differentiable parameterizations for Catmull-Clark subdivision surfaces. Unlike the natural parameterization for which derivatives diverge around extraordinary vertices of valence greater than four, the proposed reparameterizations guarantee convergence. Moreover, our method based on the inversion of the characteristic map also guarantees a parameterization with non-degenerate Jacobian, both for valences greater than four and for valence three.

At the same time, the non-degenerate parameterization carries a significant computational penalty. Ideally it would be preferable to have an approximation to the characteristic map for which the inverse can be computed exactly. A number of mathematical considerations indicate that existence of such map is unlikely if we require non-zero derivatives at the origin. At the same time, a statement of this type would be hard to prove rigorously.

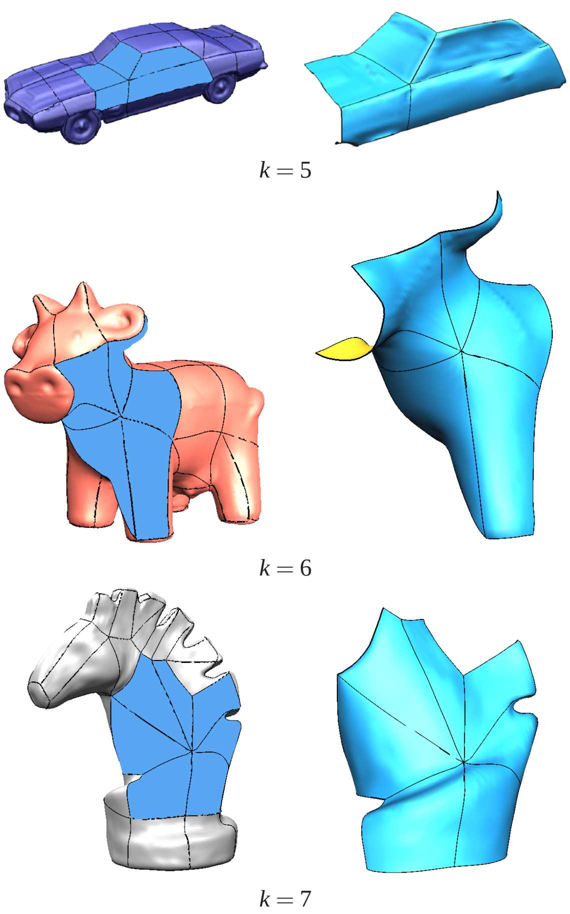


Figure 6: *Evaluation of multiresolution subdivision surfaces corresponding to scanned objects. Left column: surfaces with patches to be evaluated shown highlighted. Right column: result of evaluation after reparameterization with non-degenerate Jacobian.*

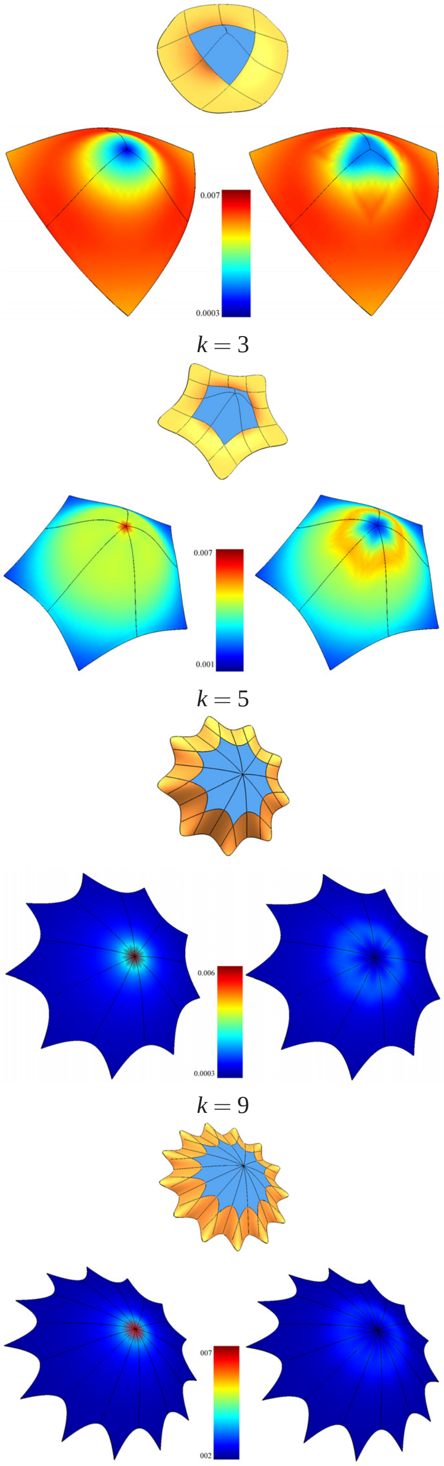
These difficulties indicate that this approach of integrating subdivision surfaces into systems designed for parametric patches has limitations. However, we believe that the approaches of this type are currently the only ones that make such integration practical for CAD applications.

Acknowledgments

This work was performed as part of a collaborative project with Dassault Systèmes. We are grateful to David Bonner and Loic LeFeuvre for their support and feedback regarding our reparameterization methods.

References

[Apo70] T. M. Apostol. Resultants of cyclotomic polynomials. In *Proc. Amer. Math. Soc.*, pages 457–



$k = 3$

$k = 5$

$k = 9$

$k = 13$

Figure 7: *Surfaces with extraordinary vertices of various valences. Top: surfaces with patches to be evaluated highlighted. Results of evaluation using the natural parameterization (left) and the reparameterization by characteristic map inversion (right) are shown color coded for comparison. The color represents the magnitude Jacobian computed based on the underlying parameterization.*

462, 1970. 6

[BFJP87] R. E. Barnhill, G. Farin, M. Jordan, and B. R. Piper. Surface/surface intersection. *Computer Aided Geometric Design*, 4(1-2):3–16, July 1987. 1

[BHLH88] C.L. Bajaj, C.M. Hoffmann, R.E. Lynch, and J.E.H. Hopcroft. Tracing surface intersections. *Computer-Aided Geometric Design*, 5:285–307, 1988. 1

[BLZ00] H. Biermann, A. Levin, and D. Zorin. Piecewise smooth subdivision surfaces with normal control. In *Proceedings of ACM SIGGRAPH 2000*, Computer Graphics Proceedings, Annual Conference Series, pages 113–120, July 2000. 1

[CC78] E. Catmull and J. Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. *CAD*, 10(6):350–355, 1978. 1

[Cd80] S. D. Conte and C. deBoor. *Elementary Numerical Analysis: An Algorithmic Approach*. McGraw-Hill, 1980. 6

[GKZ96] I.M. Gelfand, M.M. Kapranov, and A.V. Zelevinsky. *Discriminants, Resultants, and Multidimensional Determinants: Theory and Applications*. Birkhauser Boston, 1996. 6

[GN01] D. Gonsor and M. Neamtu. Can subdivision be useful for geometric modeling applications? Technical report, Boeing Corporation, 2001. 1

[GS01] E. Grinspun and P. Schröder. Normal bounds for subdivision-surface interference detection. In *IEEE Visualization 2001*, pages 333–340, October 2001. 1

[Hoh91] M. E. Hohmeyer. A surface intersection algorithm based on loop detection. In *SMA '91: Proceedings of the First Symposium on Solid Modeling Foundations and CAD/CAM Applications*, pages 197–208, June 1991. 1

[KP03] P. G. Kry and D. K. Pai. Continuous contact simulation for smooth surfaces. *ACM Transactions on Graphics*, 2003. to appear. 1

[KPW92] G.A. Kriezis, N.M. Patrikalakis, and F.-E. Wolter. Topological and differential-equation methods for surface intersections. *Computer Aided Design*, 24:41–55, 1992. 1

[KS99] A. Khodakovsky and P. Schröder. Fine level feature editing for subdivision surfaces. In *Proceedings of ACM Solid Modeling*, 1999. 1

[LLS01] N. Litke, A. Levin, and P. Schröder. Trimming for subdivision surfaces. *Computer Aided Geometric Design*, 18(5):463–481, June 2001. 1

[Loo87] C. Loop. Smooth subdivision surfaces based on triangles. Master's thesis, University of Utah, Department of Mathematics, 1987. 1

[PR98] J. Peters and U. Reif. Analysis of algorithms generalizing *B*-spline subdivision. *SIAM J. Numer. Anal.*, 35(2):728–748 (electronic), 1998. 4

[Rei95] U. Reif. A unified approach to subdivision algorithms near extraordinary vertices. *Comput. Aided Geom. Design*, 12(2):153–174, 1995. 5

[SC95] T. W. Sederberg and F. Chen. Implicitization using moving curves and surfaces. In *Proceedings of SIGGRAPH 95*, Computer Graphics Proceedings, Annual Conference Series, pages 301 – 308, August 1995. 6

[Sta98] J. Stam. Exact evaluation of catmull-clark subdivision surfaces at arbitrary parameter values. In *Proceedings of SIGGRAPH 98*, Computer Graphics Proceedings, Annual Conference Series, pages 395–404, July 1998. 1, 2, 3

[Sta99] J. Stam. Evaluation of loop subdivision surfaces. SIGGRAPH'99 Course Notes, 1999. 1

[ZK02] D. Zorin and D. Kristjansson. Evaluation of piecewise smooth subdivision surfaces. *The Visual Computer*, 18(5/6):299–315, 2002. 1

Appendix A: Proofs

Proof of Proposition 1. If the Jacobian of the map does not vanish away from the vertices, the map is a covering on its image (see any basic topology text). As $[0,1]^2 \setminus \{v_1, v_2, v_3, v_4\}$ is simply connected, the covering has to be an isomorphism. It is easy to check directly that it leaves the vertices v_i in place. This means that F_b is one-to-one. As it is continuous, it maps the boundary of $[0,1]^2$ to the boundary of the image. It can be easily verified that F_b maps each side of the square $[0,1]^2$ to itself. Consider the half of the edge $e = [v_i, v_{i+1}]$ adjacent to v_i. F_i maps the half-line obtained by extending v_i, v_{i+1} beyond v_{i+1} to itself. As F_b on e is an affine combination of F_i and identity, it maps e to the straight line containing e. The Jacobian in the interior of e does not vanish, therefore, the derivative of the map $F_b|_e$ along e has constant sign in the interior of e. The function is monotonic in the interior and continuous, therefore monotonic on the whole e. As it maps the endpoints of e to themselves, it follows that e is mapped to e. We conclude that the boundary of $[0,1]^2$ is mapped onto itself and, therefore, the interior is mapped to the interior. Finally, if there is a point in the interior of $[0,1]^2$ which is not an image of any point, then there is a curve in $F_b([0,1]^2)$ which is not contractible, which is impossible given that the image is homeomorphic to $[0,1]^2$. We conclude that in the interior the map is also onto.

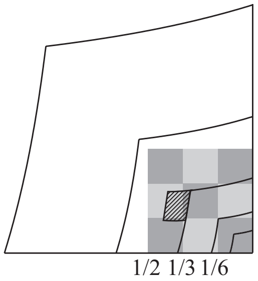


Figure 8: *The boundaries of layers of the characteristic maps are shown together with the domains on which w is polynomial (shades of gray). A single subpatch of the characteristic map is hatched. We verify that the Jacobian of the composition does not vanish for the subpatch of the characteristic map composed with any of the overlapping polynomial pieces of w.*

Proof of injectivity of the characteristic map reparameterization. According to Proposition 1 it is sufficient to show that the Jacobian of the map does not vanish. It is sufficient to consider the map restricted to one quarter of the domain adjacent to one of the corner vertices. On this subdomain the map is a blend between the identity and the inverse of the characteristic map: $w\Phi^{-1} + (1-w)Id$. We note that proving that the Jacobian for this map does not vanish is equivalent to proving that the Jacobian of $(w \circ \Phi)Id + (1 - w \circ \Phi)\Phi$ does not vanish.

The proof of the latter is straightforward, but requires considerable amount of algebraic calculations which we have performed using a computer algebra system. We outline here the main steps of the analysis.

Assume that the valence is fixed. The computation below needs to be performed for all valences.

First, we recall that the characteristic map is piecewise-polynomial and that the polynomial subpatches have a layered structure near the extraordinary vertex. Furthrmore, the map is self-similar, so there are only three basic patches $P_j(u,v)$, $j = 1,2,3$. All the rest have the form $\lambda^i P_j(2^{-i}u, 2^{-i}v)$.

As w is also piecewise polynomial, the combined map is piecewise polynomial. Unfortunately, the shape of the polynomial patches is not simple, as some of the boundaries occur, for example, when $\Phi(u,v)$ is on the line segment $e = (1/c, t), 0 < t < 1/c$ at which w switches from one polynomial form to another. This is illustrated in Figure 8.

This means that the patch boundary in the (u,v) domain is the curve $\Phi(u,v)$ which makes it difficult to verify exactly that the Jacobian is positive.

We take the following approach. The domain is subdivided N times (we use $N = 3$), and each of $2^N \times 2^N$ subpatches is examined separately. The value of N is chosen such that for the subpatch $(0,0)$, i.e., for $u,v < 2^{-N}$ $\Phi(u,v) < 1/c$. This implies that $w(\Phi(u,v)) = P_0^w(\Phi(u,v)$, where P_0^w is a fixed polynomial.

For all other patches (l,m), $l \neq 0$ or $m \neq 0$, we observe that $\Phi(u,v)$ reduces to a polynomial. We convert each patch to Bezier form and we use the bounding box of the Bezier control points to determine the range of values of $\Phi(u,v)$. Using these ranges we determine which polynomial pieces of w are necessary. If the patch (l,m) of the characteristic map overlaps a line separating two pieces of w, we include both pieces. As a result, for each subpatch we have a set of polynomial forms of the reparameterization map. These forms correspond to the smaller subdomains on which $(w \circ \Phi)Id + (1 - w \circ \Phi)\Phi$ is polynomial.

Next, we compute the Jacobian of each patch $(l,m) \neq (0,0)$, which is a polynomial in (u,v) and we convert it to Bezier form. We verify that the coefficients of the Bezier form have the same sign. By the convex hull property, this proves that the Jacobian does not vanish anywhere, except possible on the patch $(0,0)$.

In the subdomain $(0,0)$ there is an infinite number of polynomial subpatches. Fortunately the polynomials on all these subpatches have one of three possible general forms, due to the fact that w reduces to a polynomial on this subpatch and to the self-similarity of the characteristic map. Specifically, the patches have the form:

$$2^{-i}P_0^w(\lambda^i P_j(u,v))Id + \lambda^i(1 - P_0^w(\lambda^i P_j(u,v)))P_j(u,v)$$

for $0 \leq u,v \leq 1$.

We use the following approach to verify that the Jacobian of this map is positive. We formally replace $2^{-(i-N)}$ with a new variable z, (N is subtracted to ensure that $z \in [0,1]$) and similarly for λ^{i-N}. This results in the expression:

$$2^{-N}zP_0^w(\lambda^N tP_j(u,v))Id + \lambda^N t(1 - P_0^w(\lambda^N tP_j(u,v)))P_j(u,v)$$

which is a polynomial in four variables z,t,u,v, all changing in the range 0 to 1. We can verify that the values of the Jacobian of this map are positive for all values of the variables converting to Bezier form. Note that continuous ranges z and t include all discrete values we need to verify positivity for.

This procedure is repeated for each valence. We note that one needs to use interval arithmetic for Bezier coefficients as explicit closed form expressions are rather complex. Finally, for large values of k, the control meshes of the characteristic map patches converge to a limit. Thus, using interval arithmetic, the test can be performed for all values of $k > k_0$ simultaneously, with k_0 chosen sufficiently large so that $\cos(2\pi/k_0)$ is close to 1. We choose $k_0 = 40$.