CURVATURE-CONTINUOUS BICUBIC SUBDIVISION SURFACES FOR POLAR CONFIGURATIONS

By

ASHISH MYLES

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2008

To the betterment of education everywhere

ACKNOWLEDGMENTS

Thanks go to my supervisory committee and everyone involved in SurfLab, especially my advisor Jörg Peters, for giving me so much intuition and positive experience during my 8 years working in graphics and geometry. I additionally thank Xiaobin Wu for helping me develop my early intuition on subdivision surfaces during our internship at ATI, and my roommate Il (Memming) Park for our math brainstorm, which improved my intuition of the theoretical structure underlying this work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

$\sum_h^n$, $\sum_\eta^S$	equivalent to $\sum_{h=0}^{n-1}$ for positive integers n and $\sum_{\eta\in S}$ for sets S, respectively
A	block-circulant subdivision matrix
$\hat{A}$	block-diagonalization of subdivision matrix A computed via Fourier transform
$\hat{A}_k$	k^{th} Fourier block of $\hat{A}$
α_k	index of the Fourier block $\hat{A}_{\alpha_k}$ which contributes the eigenvalue ℓ_k
c_τ, $c_{j:n}$	$\cos(2\pi\tau)$, $\cos\left(2\pi\frac{j}{n}\right)$
e_k	eigenspline
$\hat{e}_k$	Fourier radial eigenspline
$\mathcal{F}$	Fourier block matrix with 6×6 blocks.
$\mathcal{G}$	operator that, when applied to a polar configuration $\mathbf{q}$, returns the set of all its circular Greville abscissae $\mathcal{G}\mathbf{q} = \left\{\frac{j}{n}\right\}_{j\in\mathbb{Z}_n}$
G_m	operator that converts a polar configuration $\mathbf{q}^m$ to the uniform periodic cubic spline ring $G_m\mathbf{q}^m : [2\lambda^m, 4\lambda^m]\times\mathbb{R}_1 \to \mathbb{R}$ (see Figure 3-2, pp. 28) $$G_m\mathbf{q}^m(r,\tau) := \sum_{i=1}^{5}\sum_{j}^{n}\mathbf{q}_{ij}^m \overset{*}{N}_i^{(m)}(r)\overset{\circ}{N}_j^{(n_m)}(\tau), \qquad \lambda := \frac{1}{2}$$
$\hat{G}_m$	operator that converts a vector $\mathbf{u}\in\mathbb{R}^6$ to the uniform cubic spline $\hat{G}_m\mathbf{u} : [2\lambda^m, 4\lambda^m]\to\mathbb{R}$ $$\hat{G}_m\mathbf{u}(r) := \sum_{i=1}^{5}\mathbf{u}_i\overset{*}{N}_i^{(m)}(r).$$
ℓ_k	eigenvalue of A with the k^{th} largest modulus, counting multiplicity

Λ	diagonal matrix of the eigenvalues of A (respectively, $\hat{A}$)
$\mathcal{L}$	operator that converts a polar configuration to its limit surface in polar parameterization: $\mathcal{L}(\mathbf{q}) := \mathbf{x}$
n_m	valence of the polar vertex of the polar configuration $\mathbf{q}^m$
$\overset{*}{N}{}_i^{(m)}(r)$	the i^{th} uniform cubic B-spline basis with knots $\frac{1}{2^m}[-1, 0, 1, 2, 3, 4, 5, 6]$
$\overset{\circ}{N}{}_j^{(n)}(\tau)$	the j^{th} uniform periodic cubic B-spline basis with knots $\frac{1}{n}\mathcal{Z}_n$
$\text{op}_k(\eta)$	$\begin{cases} c_{\alpha_k \eta} & \text{if } k \leq n/2 \\ -s_{\alpha_k \eta} & \text{otherwise} \end{cases}$
$\mathbf{p}_k$	eigencoefficient corresponding to eigenspline e_k
$\mathbf{q}^m$	an m-times refined polar configuration
$\mathbb{R}$, $\mathbb{R}_1$	the set of reals and the set of reals modulo 1, respectively
s_τ, $s_{j:n}$	$\sin(2\pi\tau)$, $\sin\left(2\pi\frac{j}{n}\right)$
$V_{\mathbb{R}}$ (resp. $V_{\mathbb{C}}$)	matrix whose columns are the real (respectively, complex) right eigenvectors of A
$\hat{V}$	matrix whose columns are the right eigenvectors of $\hat{A}$
$\mathbf{v}_k$, $\hat{\mathbf{v}}_k$	right eigenvector of A and A_{α_k}, respectively, corresponding to ℓ_k
$\mathbf{w}_k$, $\hat{\mathbf{w}}_k$	left eigenvectors of A and A_{α_k}, respectively, corresponding to ℓ_k
$\mathbf{x}$	polar limit surface in polar parameterization
$\overline{\mathbf{x}}$	polar limit surface in Cartesian parametrization
$\mathbb{Z}$, $\mathbb{Z}_n$	the set of integers and the set of integers modulo n, respectively
$\mathcal{Z}_n$	the strictly increasing sequence of integers in $\mathbb{Z}_n$

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

CURVATURE-CONTINUOUS BICUBIC SUBDIVISION SURFACES FOR POLAR CONFIGURATIONS

By

Ashish Myles

December 2008

Chair: Jörg Peters
Major: Computer Engineering

Subdivision surfaces are popular in animation as a way of smoothing coarse control meshes. On the other hand, the Computer-Aided Design (CAD) industry typically prefers the simplicity and predictability of NURBS when constructing high-quality surfaces for the manufacture of cars and planes. Since a single NURBS patch is capable only of modeling the topologies of planes, cylinders, and torii, it is complex to use a NURBS atlas to construct a surface of arbitrary topology that is curvature-continuous everywhere. While popular subdivision algorithms of low parametric degree, like Catmull-Clark and Loop subdivision, are not inherently restricted in topology, they suffer from shape artifacts at so-called "extraordinary vertices". This makes them unattractive for CAD. Subdivision theory requires a (bi)degree of at least 6 in order for stationary subdivision to be non-trivially curvature-continuous and mitigate some of these shape artifacts. We circumvent this restriction by designing a curvature-continuous non-stationary bicubic subdivision algorithm which has the implementational simplicity of stationary algorithms. We hope techniques such as ours make subdivision surfaces more attractive for high-quality constructions in CAD.

CHAPTER 1
INTRODUCTION

From automobile and plane design to digital movie animation to video and computer game character design, smooth curves and surfaces play a fundamental role in the design of objects. Standard Computer-Aided Design (*CAD*) packages need to represent these surfaces in an efficient form that is easy to manipulate algorithmically, and intuitive for the user to mold into the desired shape. Additionally, such surface representations should be easy to visualize and render onto the screen.

Smooth surface representations in CAD packages can be largely classified into two categories: implicit and parametric. Implicit surfaces are defined in terms of zero-sets. For example, $x^2 + y^2 + z^2 - 1 = 0$ is the implicit representation of the unit sphere. While this representation is useful to create basic shape and to apply boolean operations, visualizing and rendering the surface typically requires solving a set of non-linear equations.

The alternative is to use parametric representations. In contrast to its implicit form, a unit sphere can be represented using three equations in terms of two parameters s and t as follows.

$$x(s,t) = \cos(s)\cos(t), \qquad y(s,t) = \cos(s)\sin(t), \qquad z(s,t) = \sin(s)$$

As s is varied from 0 to π and t is varied from 0 to 2π, the points on the surface of the

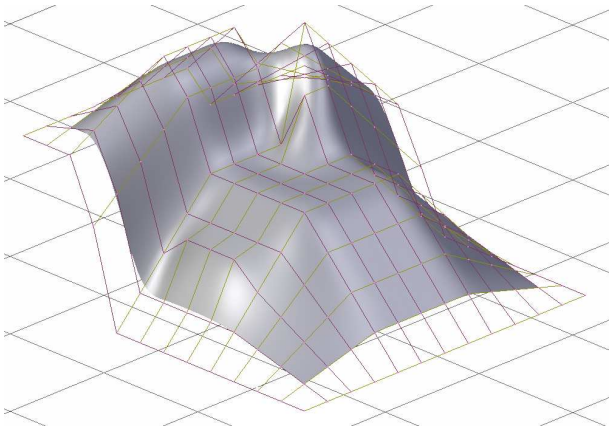


Figure 1-1. A NURBS surface in a typical CAD package determined by a control net consisting of all quads and internal vertices of valence 4.

sphere are generated. A standard form of parametric representation is called *Non-Uniform Rational B-Splines*, or NURBS for short. In this case, the x-, y-, and z-coordinates are represented separately by piecewise rationals. The surface is defined by A) a *control mesh*, or control net, which for NURBS is a quad mesh with each internal vertex having valence 4, as shown in Figure 1-1; B) two *knot sequences* that determine the extent and effect of the domain, one for each of the s and t parameters; and, C) *weights* associated with each vertex in the control mesh. Even though boolean operations on NURBS are not straightforward, NURBS are popular due to their intuitive manipulability and ease of rendering. However, being based on quad-grid control structures, NURBS are capable of representing only topological planes, cylinders, or torii. While the theory of NURBS will not be discussed, a special case of it is important in this study: the *uniform B-spline*, with uniformly spaced knots and all weights 1. Surfaces in B-spline form can be converted directly to closed form, which is useful for analysis. Alternatively, the surface can be defined via an iterative mesh refinement algorithm, which is easier to generalize. Chapter 2 discusses uniform bi-degree-3 splines in greater detail.

Table 1-1. Various mesh refinement algorithms (not comprehensive). Quad/triangle is only C^1 over certain edges and isolated points. Except for TURBS, all produced surfaces are generically only C^1 at isolated points. The last column indicates whether or not the algorithm interpolates its control points.

Year	Algorithm	Smooth	Degree	Basis	Interp.
1978	Catmull-Clark [Catmull and Clark, 1978]	C^2	bi-3	□	no
1978	Doo-Sabin [Doo and Sabin, 1978]	C^2	bi-2	□	no
1987	Loop [Loop, 1987]	C^2	4	△	no
1990	Butterfly [Dyn et al., 1990]	C^1	N/A	△	yes
1996	Kobbelt [Kobbelt, 1996]	C^1	N/A	□	yes
1997	Simplest [Peters and Reif, 1997]	C^1	2	□	no
1998	TURBS [Reif, 1998]	C^k	bi-$(2k+2)$	□	no
2000	$\sqrt{3}$ [Kobbelt, 2000]	C^2	N/A	△	no
2001	4–8 [Velho and Zorin, 2001]	C^4	6	□	no
2001	Circle preserving [Morin et al., 2001]	C^2	3 & trig.	□	no
2002	Ternary triangle [Loop, 2002b]	C^4	4	△	no
2003	Quad/triangle [Stam and Loop, 2003]	C^2	bi-3, 4	△, □	no
2004	4–3 [Peters and Shiue, 2004]	C^2	4	△, □	no

To address the inherent limitations of NURBS, subdivision surfaces were introduced simultaneously by Catmull and Clark [1978] and Doo and Sabin [1978]. The two subdivision surface algorithms are generalizations of B-spline iterative mesh refinement rules, supporting arbitrary connectivity and manifold topology. These rules specify where points are added; how the positions of these points are computed; and how the mesh is reconnected. After an infinite iterative application of these subdivision rules, the mesh converges to a limit surface. Some subdivision algorithms were created specifically for triangular meshes, whereas others were created for quad meshes. Some were created specifically to interpolate the vertices of the control net. Some were designed for tangent continuity (C^1), and others for curvature continuity (C^2). Table 1-1 summarizes several well-known subdivision algorithms, and is by no means complete. The surface quality of various C^2 algorithms listed is deficient at certain isolated points, called *extraordinary points*, where they are only C^1. What is considered to be an extraordinary point depends on the details of each algorithm. Section 2.2, for instance, will define the extraordinary point for Catmull-Clark surfaces and describe the surface behavior in its neighborhood. The literature on the analysis techniques is enumerated at the end of Section 2.2.4.

Various surface construction algorithms were invented or adapted for applicability or quality. For example, quad/triangle subdivision mentioned in Table 1-1 is a combination of Catmull-Clark and Loop subdivisions applied to the quad and triangular portions of the mesh separately. New rules were developed for the boundary between the quad and triangle meshes, and the behavior of the surface along those edges is only C^1. Since Catmull-Clark by itself was designed for quads and has undesirable shape on triangle meshes, its combination with Loop's algorithm improves overall surface quality and the applicability of the subdivision algorithm. Addressing surfaces of revolution, Morin et al. [2001] designed a subdivision algorithm capable of reproducing circles, which polynomial algorithms cannot do. This technique reproduces cubic polynomials, circles, and hyperbolic functions depending on a tension parameter. By tensoring the algorithm

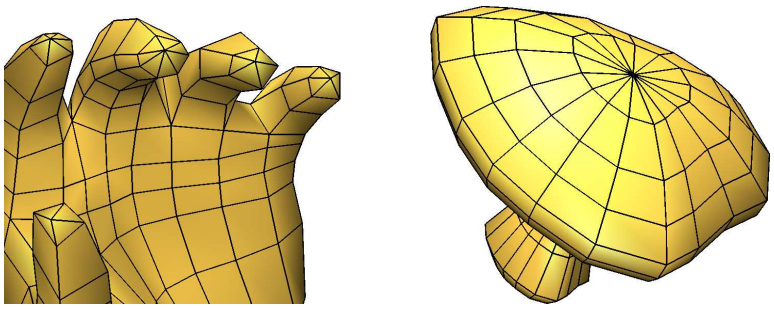
on a quad mesh, they obtained a surface that is C^2 except at extraordinary points, where it is C^1. While most subdivision algorithms approximately quadruple the number of points in the mesh after every refinement, some are specifically designed to refine slowly: simplest subdivision [Peters and Reif, 1997] and 4–8 subdivision [Velho and Zorin, 2001] quadruple every two iterations; $\sqrt{3}$ subdivision [Kobbelt, 2000] increases 9-fold every two iterations. Slowing the refinement gives greater control over the size of the refined mesh. This is useful for rendering no more than is necessary.

Catmull-Clark and Loop subdivision, the most well-known subdivision algorithms for quad and triangle meshes, respectively, are known to have unbounded curvature in the vicinity of the extraordinary point. Many attempts have been made to improve upon them. Sabin [1991] re-tuned Catmull-Clark so that it yielded surfaces with bounded curvature. Augsdörfer et al. [2006] went a step further to minimize Gaussian curvature variation within the space of bounded curvature algorithms. Various modifications have been made to Loop subdivision to support curvature continuity, albeit with a local flat spot with zero curvature [Prautzsch and Umlauf, 1998, 2000]; bounded curvature with the surface lying within the convex hull of the control points [Loop, 2002a,b]; and, curvature control [Ginkel and Umlauf, 2006]. Umlauf [2005] summarized many of these re-tuning techniques.

Notable constructions that support arbitrary degree of smoothness even at the extraordinary point include free-form splines [Prautzsch, 1997] and TURBS [Reif, 1998], both of which require degree bi-$(2k + 2)$ to create an everywhere-C^k surface. Ying and Zorin [2004] created an everywhere-C^∞ surface using exponential blending functions between polynomial patches. More recent work by Karčiauskas and Peters [2007b, 2008] introduced the concept of guided subdivision also capable of achieving arbitrary continuity. For C^2, they employ an infinite sequence of bi-degree-6 spline surface rings to approximate a C^2 "guide surface" of good quality. In [Karčiauskas and Peters, 2007d], they employ sequences of bicubic spline rings containing an exponentially-increasing

number of polynomials to reproduce the guide surface's second order behavior at the extraordinary point in spite of the low degree of the overall construction. Our construction implicitly also uses bicubic spline surface rings of exponentially-increasing number of polynomials to achieve curvature continuity; however, this increase comes about naturally in our algorithm.

A variety of other approaches have been used to improve shape near extraordinary points. Peters [2000] approximated Catmull-Clark surfaces with a finite number of bicubic patches that join tangent-continuously. As an alternative, Peters [2002] suggested a C^2 construction of degree $(3, 5)$. Both these techniques still suffer from shape problems due to the low degree of the constructions. Loop and Schaefer [2008] achieved curvature continuity for quad meshes using patches of bi-degree 7 with shape optimization for the free parameters. Karčiauskas and Peters [2007c] used the concepts of guided subdivision to construct a C^2 surface with a finite number of bi-degree-6 patches. [Levin, 2006] perturbed Catmull-Clark surfaces using polynomial-square-root blending functions between local polynomial patches. In the same vein, Zorin [2006] perturbed Loop subdivision surfaces to be C^2 using a blending function that was itself a subdivision surface.



A B

Figure 1-2. Polar configuration on A) finger tips and B) the top of the mushroom.

Many of the surface construction algorithms mentioned above are complex or suffer in shape near high-valence vertices. Karčiauskas and Peters [2007] recognized one commonly-occurring configuration of high valence in quad-dominant meshes: *the polar configuration*, which is the focus of this study. The polar configuration consists

of a high-valence central vertex – the *polar vertex* – in the middle of a triangle fan surrounded by a quad grid neighborhood. This configuration occurs naturally at the ends of elongated objects like tips of fingers, and in the latitude/longitude connectivity of the sphere (Figure 1-2), and it is structurally far simpler than the neighborhood of Catmull-Clark extraordinary points, as we show in Section 2. Catmull-Clark on polar configurations results in macroscopic oscillations in the polar neighborhood. Treating polar as a special case gives good results, even when the central valence is very high (Figure 2-6). Karčiauskas et al. [2006] adapted guided subdivision to polar configuration to create C^2 polar jet subdivision, which employs a control net structure to make spline surface rings of degree $(6, 5)$. Karčiauskas and Peters [2007a] introduced very simple bicubic C^1 subdivision algorithm with bounded curvature, which was subsequently adapted by Myles et al. [2008] to be compatible with Catmull-Clark subdivision. Myles et al. [2008] also offered a C^1 bicubic NURBS patch construction with bounded curvature to cover the neighborhood of the polar configuration.

There is no accepted mathematical definition of surface quality. For simulation, it is often useful to have well-defined curvatures. Additionally, the introduction of curvature continuity tends to improve visual quality of the modeled surface. Subdivision theory [Peters and Reif, 2008] states that Catmull-Clark subdivision cannot be re-tuned to be non-trivially C^2 at the extraordinary point with degree less than bi-6. In this study, we sidestep the assumptions underlying this theorem to take advantage of the natural subdivision structure of polar configurations to create a C^2 algorithm that has degree only bi-3. We also show that our simple subdivision algorithm yields surfaces with high visual quality and good curvature distribution.

CHAPTER 2
GENERALIZATIONS OF UNIFORM BICUBIC SPLINES

2.1 Uniform B-Spline Representation

We introduce notation and definitions to simplify the discussion.

- For integers n, $\sum_h^n$ is an alternative notation for $\sum_{h=0}^{n-1}$. For sets S, $\sum_\eta^S$ is an alternative notation for $\sum_{\eta \in S}$.
- $\mathbb{Z}$ is the set of integers, and $\mathbb{Z}_n$ is the integers modulo n. $\mathbb{R}$ is the set of reals. $\mathbb{R}_1$ is $\mathbb{R}$ modulo 1. $\mathcal{Z}_n$ is the strictly increasing sequence of integers in $\mathbb{Z}_n$.
- An *affine combination* is a linear combination where the weights add to 1. A *convex combination* is an affine combination where the weights are positive.

2.1.1 Univariate

A detailed treatment of the B-spline form can be found in [Prautzsch et al., 2002]. A piecewise polynomial in B-spline form is defined by a sequence of *control points* that defines the shape, and a uniformly-spaced knot sequence that defines the domain. The piecewise linear interpolant for a given ordering of control points is known as the **control polygon** (see Figure 2-1). A univariate cubic (i.e. degree 3) uniform spline $f : \mathbb{R} \to \mathbb{R}$ with n control points $\mathbf{b} := [b_0, b_1, \ldots, b_{n-1}]$ requires $n + 4$ uniformly-spaced knots $\mathbf{t} := [t_0, t_1, \ldots, t_{n+3}]$ and is defined by

$$f(t) := \sum_i^n b_i N_i(t),$$

where the n cubic B-spline bases $N_i(t)$ are

$$N_i(t) := \frac{1}{6} \begin{cases} u_i(t)^3 & \text{if } t \in [t_i, t_{i+1}] \\ -3u_{i+1}(t)^3 + 3u_{i+1}(t)^2 + 3u_{i+1}(t) + 1 & \text{if } t \in [t_{i+1}, t_{i+2}] \\ 3u_{i+2}(t)^3 - 6u_{i+2}(t)^2 + 4 & \text{if } t \in [t_{i+2}, t_{i+3}] \\ (1 - u_{i+3}(t))^3 & \text{if } t \in [t_{i+3}, t_{i+4}] \\ 0 & \text{otherwise} \end{cases} \qquad u_i(t) := \frac{t - t_i}{t_{i+1} - t_i}. \tag{2–1}$$

While the spline is technically defined on all of $\mathbb{R}$, it is restricted for practical purposes

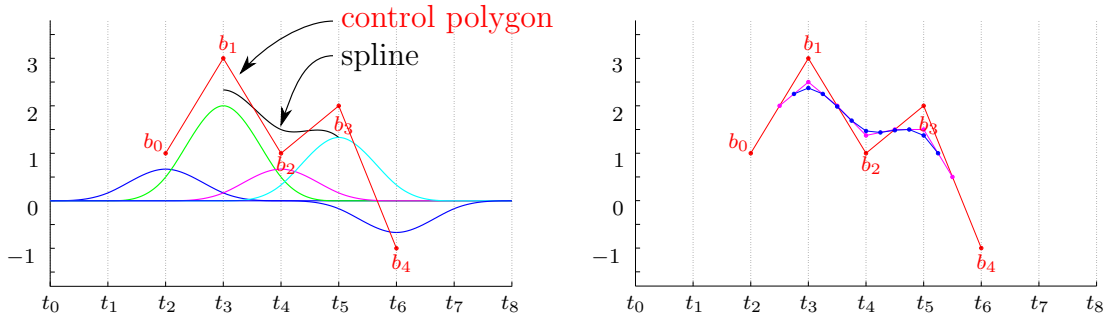


Figure 2-1. Univariate uniform cubic spline. A) A cubic spline $f(t)$ with control points $\mathbf{b} = [1, 3, 1, 2, -1]$ (*red*) and knots $\mathbf{t} = [-1, 0, 1, 2, 3, 4, 5, 6, 7]$ is the sum of uniform B-spline bases scaled by their respective control points (*blue, green, magenta, cyan*). B) An equivalent definition using iterative control polygon refinement.

to $t \in [t_3, t_n]$, as in Figure 2-1, where at least four non-zero bases overlap. The basis functions are non-negative and sum to one in this interval, implying that each point on the spline is a convex combination of the control points. This yields two important geometric properties of B-splines.

- **Affine invariance:** Applying an affine transformation to the control polyhedron applies it to the transformation spline as well.
- **Convex hull property:** A parametric curve in B-spline form always lies in the convex hull of its control points.

Uniform cubic B-splines also have built-in second-order continuity so that adjacent polynomial pieces join C^2.

The t-coordinate associated with each control point $\mathbf{b}_i$ is called the *Greville abscissa* t_i^* and is defined, in general, via $t_i^* := \frac{1}{d}\sum_j^d t_{i+j+1}$, where d is the degree of the spline. For uniform cubics, this simplifies to $t_i^* = t_{i+2}$. It will be useful later to index control points by their Greville abscissae when the knot sequence is chosen so that $t_i^* = \frac{i}{n}$. To this end, we define the operator $\mathcal{G}$

$$\mathcal{G}\mathbf{b} := \{t_i^*\}_{i \in \mathbb{Z}_n}, \tag{2–2}$$

and the bracketed fractional indexing notation.

$$b_{\left[t_i^*\right]} = b_{\left[\frac{i}{n}\right]} := b_i, \quad N_{\left[t_i^*\right]}(t) = N_{\left[\frac{i}{n}\right]}(t) := N_i(t). \tag{2–3}$$

Using this notation, our spline is equivalently defined as

$$f(t) := \sum_{\eta}^{\mathcal{G}\mathbf{b}} b_{[\eta]} N_{[\eta]}(t).$$

One can similarly define periodic uniform cubics $f : \mathbb{R}_1 \to \mathbb{R}$, requiring the knot sequence to lie within $\mathbb{R}_1$. Since the knot sequence cycles around, we need only specify n knots – e.g. $\mathbf{t} = \frac{1}{n}\mathcal{Z}_n$ – and assume by convention that the first control point has Greville abscissa 0.

The B-spline form can alternatively be defined via a control polygon refinement procedure as illustrated in Figure 2-1B. The once-subdivided control points $\mathbf{b}^1 := [b_1^1, b_2^1, \ldots, b_{2n-3}^1]$ are computed from the original control points $\mathbf{b}^0 := [b_0^0, b_1^0, \ldots, b_{n-1}^0]$ via the following equations.

$$b_{2i}^1 = \frac{1}{8}b_{i-1}^0 + \frac{6}{8}b_i^0 + \frac{1}{8}b_{i+1}^0, \qquad b_{2i+1}^1 = \frac{1}{2}b_i^0 + \frac{1}{2}b_{i+1}^0 \tag{2–4}$$

Applying this refinement procedure ad-infinitum converges to the spline curve.

2.1.2 Tensor-product bivariate

The B-spline bases can be easily generalized to surfaces by tensoring the univariate bases, so that the *bi-3* (i.e. bicubic, bi-degree-3, or degree (3,3)), surface $f(s,t)$ is defined as

$$f(s,t) := \sum_{i}^{n_s} \sum_{j}^{n_t} b_{ij} N_i^s(s) N_j^t(t),$$

where the spline is defined by the $n_s \times n_t$ *control mesh* b of control points, and two knot sequences $\mathbf{s} := [s_0, s_1, \ldots, s_{n+3}]$ and $\mathbf{t} := [t_0, t_1, \ldots, t_{n+3}]$ which define the B-spline bases $N_i^s(s)$ and $N_j^t(t)$, respectively. The Greville abscissa of a control point b_{ij} is a pair (s_i^*, t_j^*)

instead of a single real. Figure 1-1 illustrates such a uniform bi-3 spline in a typical CAD environment.

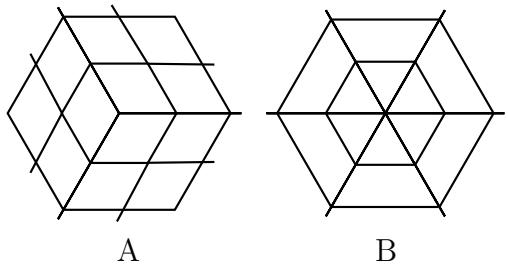


Figure 2-2. Generalizations of mesh connectivity. A) A quad-only generalization to mesh connectivity allows vertex valences other than 4. B) Polar mesh connectivity arrives naturally when many control lines along the same tensor direction meet at a singularity.

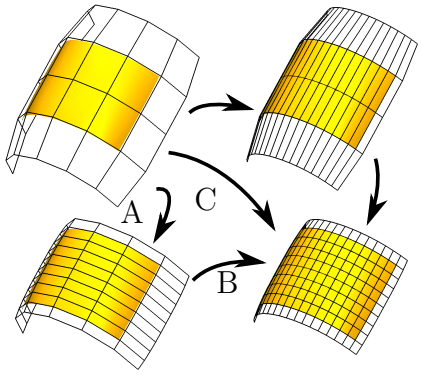


Figure 2-3. Commutativity of regular B-spline subdivision. Bi-3 spline subdivision A) in one direction followed by B) the other, or C) simultaneous refinement as in Catmull-Clark.

The surface can also be defined using a mesh refinement procedure. The control mesh may be subdivided in either tensored direction independently, to yield the same surface in closed form. Figure 2-3 illustrates this procedure on a parametric spline, where the mesh is (A) refined strictly in one direction twice followed by (B) the other one twice. Alternatively, one can tensor the subdivision masks so that one may (C) directly subdivide in both directions twice, converting each original quad into four refined ones after every tensored subdivision.

2.2 Catmull-Clark and Bi-3 Polar Subdivision

2.2.1 Generalizations of quad grid meshes

Quad-grid control meshes are limited to representing the topology of planes, cylinders, and torii. Generalizing the mesh connectivity to allow arbitrary-valence vertices and polar configurations (Figure 2-2) admits meshes encoding arbitrary surface topology. The B-spline quad-grid connectivity is called the *ordinary case*. The *extraordinary case* consists of a quad neighborhood with an *extraordinary vertex* of valence $\neq 4$, and its neighboring quads are called *extraordinary quads*. The *polar configuration*, as defined previously, consists of a central *polar vertex* of arbitrary valence ≥ 3, surrounded by a triangle fan within rings of regular quads (Figure 2-2B).

While the utility of arbitrary valences may be obvious, an appreciation for polar configurations requires more observation. Figure 1-2 already demonstrates their utility on certain meshes, but we will justify treating polar configurations specially when examining subdivision surfaces in the following section.

2.2.2 Subdivision as refinement operations

The Catmull-Clark subdivision algorithm Catmull and Clark [1978] takes an arbitrary input mesh and subdivides it to produce a denser mesh on which the algorithm can again be applied. The limit surface corresponding to this sequence of evermore refined meshes is called the Catmull-Clark (limit) surface. For simplicity of discussion, we will assume closed meshes – i.e. those with no boundaries.

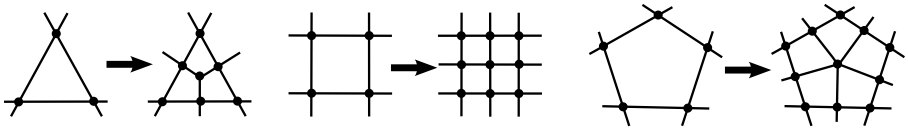


Figure 2-4. One step of Catmull-Clark refinement converts a mesh to all quads.

An application of the Catmull-Clark subdivision procedure converts every face into multiple quads by:

1. splitting every edge into two by the introduction of a new vertex (edge vertex),

2. introducing a new vertex at the middle of the face (face vertex), and

3. connecting the face vertices to their neighboring edge vertices.

See Figure 2-4 for the end result on three different polygons. Subdivision splits a face of size n into n quads and creates an n-valent vertex at the center. Since the mesh facets are four-sided after one refinement, all subsequent subdivisions quadruple the size of the mesh. The new points are affine combinations of their neighbors and the old vertices are modified using affine combinations of their old neighbors as well. Catmull-Clark subdivision, demonstrated in Figure 2-5A, can hence be thought of (and was originally derived) as a generalization of Figure 2-3C, where both directions of the mesh are subdivided in a single step.

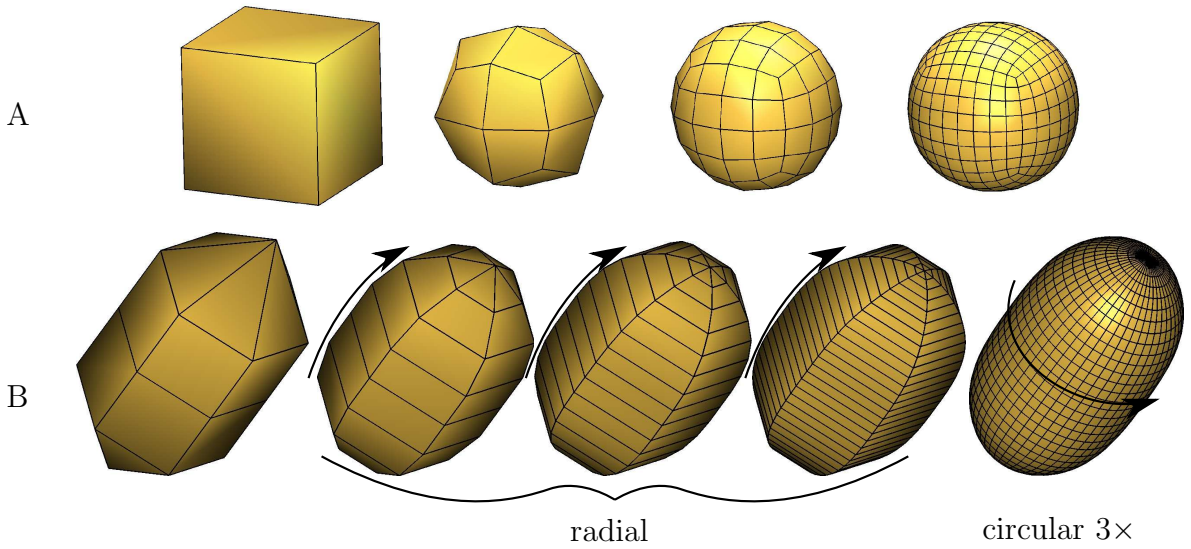


Figure 2-5. A) Catmull-Clark subdivision splits every quad directly into four, using special rules in the vicinity of extraordinary vertices, like the 3-valent ones on the cube. B) Bi-3 polar subdivision refines strictly in the radial direction the desired number of times (thrice here), and finally in the circular direction to achieve the same granularity as Catmull-Clark.

The exact refinement weights are not relevant in this discussion, and are omitted. However, it is worth noting that these weights depend only on the local connectivity of the control mesh. The weights are said to be *stationary*. Additionally, the connectivity is also *stationary*, in that the local connectivity around the extraordinary vertex has the same

structure as before – a valence n vertex surrounded by quads with all other valences being 4. Subdivision schemes with stationary weights and connectivity are themselves said to be *stationary*.

Bi-3 polar subdivision [Karčiauskas and Peters, 2007a; Myles et al., 2008] can be thought of as generalizing Figure 2-3(A-B), where subdivisions happen strictly in one direction followed by the other one. We refer to the direction along the control lines emanating radially from the polar vertex as the *radial direction*, and the periodic direction as the *circular direction*. The limit surface is defined in this case by applying subdivision in the radial direction ad infinitum, followed by subdividing in the circular direction. However, for the purposes of approximating the limit surface with the mesh, we subdivide only a finite number of times in the radial direction before we switch to the circular direction as illustrated in Figure 2-5B. Bi-3 polar subdivision is stationary, and its subdivision weights are irrelevant for this discussion. We will instead detail and analyze a slightly more complex version of this algorithm in Section 3.

Figure 2-6. Even when the polar configuration A) is convex, the Catmull-Clark subdivision surface B) has unseemly ripples, while bi-3 polar subdivision C) yields predictable surfaces.

As Figure 2-6 demonstrates, applying Catmull-Clark to a convex polar configuration results in macroscopic ripples that are clearly not specified by the control nets. However, treating polar configurations specially yields predictable surfaces with good behavior despite a high valence polar vertex. This justifies considering the polar configuration as a separate case.

2.2.3 Subdivision as piecewise polynomials

While uniform splines can be written out in closed form, Catmull-Clark and bi-3 polar subdivision surfaces are not as straightforward. As Figure 2-7 demonstrates, at any refinement level, the limit surface near extraordinary vertices and polar vertices is not directly available. However, subdividing once reveals a regular ring of quads, which undergo uniform bi-3 subdivision in subsequent refinements. The surface defined by this ring can hence be written out in closed form. Therefore, near extraordinary vertices and polar vertices, the surface consists of an *infinite* sequence of *spline surface rings* approaching a limit point. In the Catmull-Clark case, we call this limit point an *extraordinary point*, while in the polar case, it is called a *pole*.

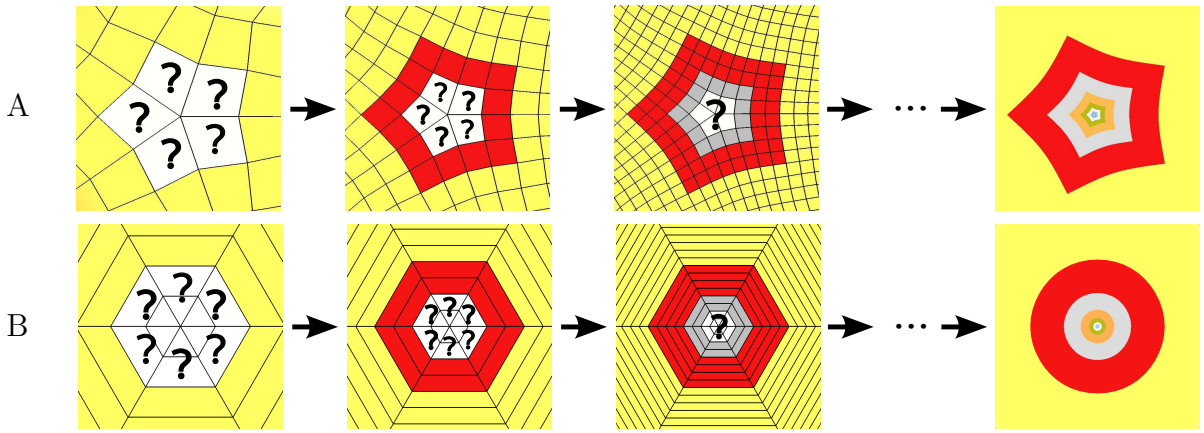


Figure 2-7. The infinite sequence of spline rings of A) Catmull-Clark and B) bi-3 polar subdivision. Each subdivision reveals additional rings of regular quads, representing bi-3 polynomial patches, around the extraordinary point.

Observe also from Figure 2-7 that the boundary of each spline ring around a Catmull-Clark extraordinary point contains as many corners as the valence of the extraordinary vertex. As the extraordinary valence approaches infinity, the boundary consists of countably infinite corners. On the other hand, in the polar case, the boundary is smooth: in a perfectly symmetric case, the boundary approaches a circle as the valence approaches infinity. We exploit this behavior of the polar configuration in the new subdivision algorithms we design.

2.2.4 Behavior around and at the extraordinary points

Since the infinite sequence of spline rings defining the surface near extraordinary points and poles are themselves C^2, the surface is C^2 away from these vertices. The behavior at the extraordinary points and poles requires a more advanced analysis of these spline sequences. Even though subdivision surfaces have been around for 30 years, only within the last 15 years were the mathematical tools for higher order analysis developed to maturity. In a landmark paper, Reif [1995] set a general framework to analyze arbitrary subdivision surface algorithms near and at the extraordinary point. Subsequent work by Prautzsch and Reif developed sufficient conditions and polynomial degrees for C^k continuity by examining the infinite sequence of spline rings [Prautzsch and Reif, 1999b,a]. Zorin [2000] instead derived conditions based on the analysis of certain "universal" surfaces that are determined by the subdivision scheme of interest. Numerous papers have analyzed the behavior of subdivision surfaces around extraordinary points [Peters and Umlauf, 2000; Sabin et al., 2003; Peters and Reif, 2004; Karčiauskas et al., 2004; Reif and Peters, 2005]. The recent book [Peters and Reif, 2008] summarizes and extends the core results of the papers above on the theory of subdivision.

Stam [1998] derived a constant-time algorithm for the evaluation of points and derivatives at parameter values arbitrarily close to the extraordinary point, and Boier-Martin and Zorin [2004] showed that a more canonical parameterization than the one used by Stam was needed to be able to always compute the derivatives at the extraordinary point.

It is now well-known that Catmull-Clark surfaces can have *unbounded curvature* near extraordinary points of valence not equal to 4 even though they are C^1 [Peters and Umlauf, 2000]. On the other hand, bi-3 polar subdivision [Karčiauskas and Peters, 2007a; Myles et al., 2008] was derived with bounded curvature in mind and tends to give predictable shapes in its areas of applicability. The purpose of this study is to go beyond bounded curvature to C^2, while still having a simple subdivision algorithm.

CHAPTER 3
RADIAL TAYLOR SUBDIVISION (RTS)

In this chapter, we will analyze *radial Taylor subdivision*, a slightly more complex variant of bi-3 polar subdivision that also has bounded curvature at the pole. Trivial modifications of radial Taylor subdivision will yield subdivision algorithms that are C^2 at the pole as well.

3.1 Notation and Labeling

The underlying data structure on which we operate is the *polar configuration* which consists of a central triangle fan surrounded by rings of ordinary quads (see Figure 3-1). The central vertex of the triangle fan is called the *polar vertex*. The *i-link* ($i = 0, 1, 2, \ldots$) of a polar configuration is the circular chain of vertices that is i edges away from the polar vertex. The 0-link consists of only the polar vertex itself. The *i-ring* ($i = 0, 1, 2, \ldots$) consists of all the vertices that are no more than i edges away from the polar vertex. While the subdivision algorithms use special rules only in the 1-ring of the polar vertex, we assume for analysis that a polar configuration constitutes the 5-ring of the polar vertex.

As illustrated in Figure 3-1, the polar configuration is denoted by $\mathbf{q}$ and its valence (i.e. the valence of its polar vertex) is n. $\mathbf{q}_i := [\mathbf{q}_{i,0}, \mathbf{q}_{i,1}, \ldots, \mathbf{q}_{i,n-1}]^{\mathrm{T}}$ denotes the i-link of the polar configuration, and $\mathbf{q}_{ij}$ is the j^{th} control point (rotating counter-clockwise, modulo n) on this i-link. For $j \in \mathbb{Z}_n$, the *j-spoke* is the vector $\mathbf{q}_{*,j} := [\mathbf{q}_{0,0}, \mathbf{q}_{1,0}, \ldots, \mathbf{q}_{5,0}]^{\mathrm{T}}$. Counting the polar vertex $\mathbf{q}_{0j} = \mathbf{q}_{00}$ repeatedly, $\mathbf{q}$ has $6n$ vertices. We can enumerate all

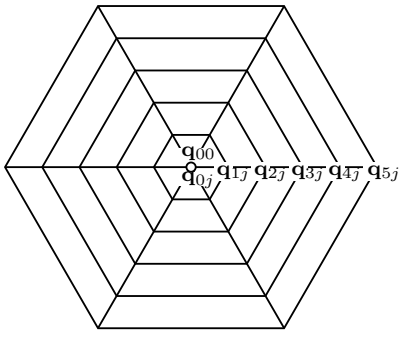


Figure 3-1. A **polar configuration** consists of a total of $6n$ control points defining the 5-ring of a polar vertex ($\mathbf{q}_{00}$, which is counted n times) of valence n.

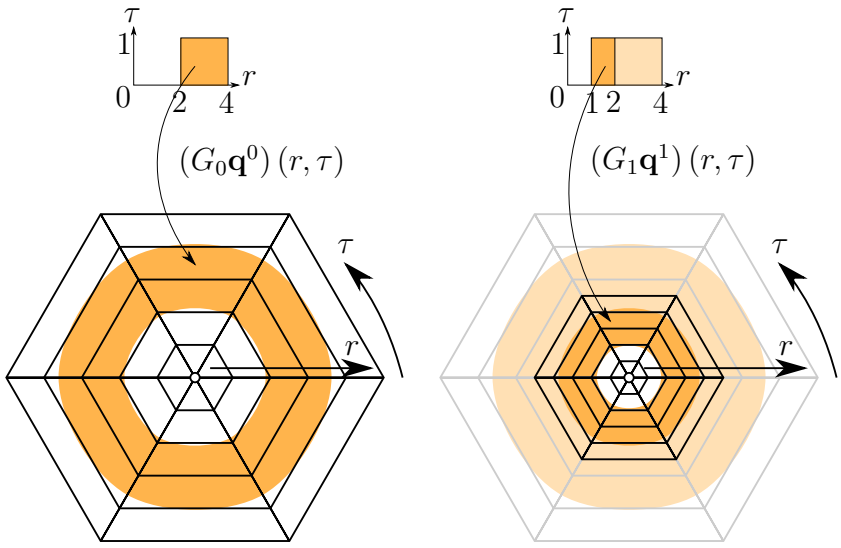


Figure 3-2. The m-times subdivided polar configuration $\mathbf{q}^m$ defines a cubic spline ring $(G_m\mathbf{q}^m)(r,\tau)$ (*orange*) via (3–1). The radial parameter shrinks by half after each subdivision so that $r \in [2,4]$ for $G_0\mathbf{q}^0(r,\tau)$ and $r \in [1,2]$ for $(G_1\mathbf{q}^1)(r,\tau)$.

these vertices as the column vector

$$\mathbf{q} := [\mathbf{q}_{0,0}, \mathbf{q}_{1,0}, \ldots, \mathbf{q}_{5,0}, \quad \mathbf{q}_{0,1}, \mathbf{q}_{1,1}, \ldots, \mathbf{q}_{5,1}, \quad \ldots, \quad \mathbf{q}_{0,n-1}, \mathbf{q}_{1,n-1}, \ldots, \mathbf{q}_{5,n-1}]^{\mathrm{T}}$$

We will extract the spline rings described in Section 2.2.3 in such a way that each control point $\mathbf{q}_{ij}$ has the Greville abscissa $\frac{j}{n}$ in the circular direction. This allows us to simplify generalizations to infinite valences and non-stationary valence by using the valence-independent fractional indexing notation introduced in (2–3): $\mathbf{q}_{i,[\frac{j}{n}]} := \mathbf{q}_{ij}$. Since j in $\mathbf{q}_{ij}$ is modulo n, τ in $\mathbf{q}_{i,[\tau]}$ is modulo 1.

While in practice, each mesh vertex lies in $\mathbb{R}^3$, our analysis is the same as if $\mathbf{q}_{ij} \in \mathbb{R}$ since subdivision works on each coordinate independently. $\mathbf{q}^m$ is the polar configuration after m subdivisions. Omission of the superscript refers to the initial data: $\mathbf{q} := \mathbf{q}^0$. n_m is the valence of $\mathbf{q}^m$, and $n := n_0$.

The limit surface of the subdivision procedure is defined by an infinite sequence of spline rings that form a decomposition of the surface (see Figures 2-7 and 3-2). Each spline ring is the periodic uniform spline defined by the five outer links of $\mathbf{q}^m$. More precisely,

- (radial) $\overset{*}{N}_i^{(m)}(r)$ is the i^{th} cubic B-spline basis with knots $\lambda^m[-1,0,1,2,3,4,5,6,7]$, where $\lambda := \frac{1}{2}$; and
- (circular) for $n \geq 3$, $\overset{\circ}{N}_j^{(n)}(\tau)$ is the j^{th} uniform periodic cubic B-spline basis with knots $\frac{1}{n}\mathcal{Z}_n$.

The *spline ring* corresponding to $\mathbf{q}^m$ is a map $G_m\mathbf{q}^m : [2\lambda^m, 4\lambda^m] \times \mathbb{R}_1 \to \mathbb{R}$, defined by the B-spline control points $\mathbf{q}_{ij}^m$, with $i \in \{1,2,3,4,5\}$ and $j \in \mathbb{Z}_n$ such that

$$\left(G_m\mathbf{q}^m\right)(r,\tau) := \sum_{i=1}^{5}\sum_{j}^{n} \mathbf{q}_{ij}^m \overset{*}{N}_i^{(m)}(r)\overset{\circ}{N}_j^{(n_m)}(\tau), \tag{3–1}$$

where $\lambda := \frac{1}{2}$, indicating that the radial parameter of each spline ring shrinks by half after every subdivision as illustrated in Figure 3-2. As $n_m \to \infty$, the i-link converges to a *curve* $\mathbf{q}_{i,[\tau]}^m$ with $\tau \in \mathbb{R}_1$, and $G_m\mathbf{q}^m$ simplifies to

$$\left(G_m\mathbf{q}^m\right)(r,\tau) := \sum_{i=1}^{5} \int_{\mathbb{R}_1} \mathbf{q}_{i,[\tau]}^m \overset{*}{N}_i^{(m)}(r)d\tau. \tag{3–2}$$

Observe also that $G_m\mathbf{q}^m$ is linear with respect to $\mathbf{q}^m$.

The *polar limit surface* $\mathbf{x} : [0,4] \times \mathbb{R}_1 \to \mathbb{R}$ ($\mathbb{R}^3$ in practice) in polar parameterization is defined piecewise in terms of these rings so that

$$\mathbf{x}(r,\tau)\big|_{r\in[2\lambda^m,4\lambda^m]} := \left(G_m\mathbf{q}^m\right)(r,\tau). \tag{3–3}$$

and $\mathbf{x}(0,\tau)$ is the unique limit point, called the *pole*. The difference between our polar parameterization and the conventional one is that our circular direction is parameterized by $\mathbb{R}_1$ instead of $\mathbb{R}_{2\pi}$ for notational convenience. The operator $\mathcal{L}$ converts a polar configuration $\mathbf{q}$ to its parameterized limit surface: $\mathcal{L}(\mathbf{q}) := \mathbf{x}$.

We propose three alternative constructions for $\mathbf{x}$ in this study that build upon each other, and show that the latter two are curvature continuous at the pole. To avoid ambiguity, we will superscript $\mathbf{x}$ and $\mathcal{L}$ by the name of the subdivision algorithm they represent – i.e. $\mathcal{L}^{\text{RTS}}(\mathbf{q}) := \mathbf{x}^{\text{RTS}}$ as defined in Section 3.2.

The operator $\hat{G}_m$ is a simpler version of the operator G_m that works strictly in the radial direction. For some vector $\mathbf{u} \in \mathbb{R}^6$, $\hat{G}_m\mathbf{u} : [2\lambda^m, 4\lambda^m] \to \mathbb{R}$ is the cubic spline defined by the B-spline control points $\mathbf{u}_i$, with $i \in \{1, 2, 3, 4, 5\}$:

$$\left(\hat{G}_m\mathbf{u}\right)(r) := \sum_{i=1}^{5} \mathbf{u}_i \overset{*}{N}{}_i^{(m)}(r). \tag{3–4}$$

Like G_m, $\hat{G}_m$ is also linear with respect to its parameter $\mathbf{u}$.

To simplify notation, we additionally define

$$c_\tau := \cos(2\pi\tau) \quad \text{and} \quad s_\tau := \sin(2\pi\tau).$$

The fraction τ can also be represented as a ratio so that

$$c_{j:n} := \cos\left(2\pi\frac{j}{n}\right) \quad \text{and} \quad s_{j:n} := \sin\left(2\pi\frac{j}{n}\right).$$

3.2 Radial Taylor Subdivision (RTS) Definition

Definition 1 (Radial Taylor subdivision)**.** Radial Taylor subdivision*, or* RTS*, refines an n-valent polar configuration* $\mathbf{q}^m$ *to the n-valent polar configuration* $\mathbf{q}^{m+1}$ *defined by*

$$\mathbf{q}_{00}^{m+1} := (1-\mathbb{a})\mathbf{q}_{00}^m + \frac{\mathbb{a}}{n}\sum_h^n \mathbf{q}_{1h}^m \tag{3–5}$$

$$\mathbf{q}_{1j}^{m+1} := (1-\hat{\mathbb{b}}_0)\mathbf{q}_{00}^m + \sum_h^n \mathbb{b}_{h-j}\mathbf{q}_{1h}^m \tag{3–6}$$

$$\mathbf{q}_{2j}^{m+1} := \mathbb{c}\mathbf{q}_{1j}^m + (1-\mathbb{c})\mathbf{q}_{2j}^m + \sum_h^n \mathbb{d}_{h-j}\mathbf{q}_{1h}^m \tag{3–7}$$

$$\mathbf{q}_{3j}^{m+1} := \frac{1}{2}\mathbf{q}_{1j}^m + \frac{1}{2}\mathbf{q}_{2j}^m \qquad \mathbf{q}_{4j}^{m+1} := \frac{1}{8}\mathbf{q}_{1j}^m + \frac{6}{8}\mathbf{q}_{2j}^m + \frac{1}{8}\mathbf{q}_{3j}^m$$

$$\mathbf{q}_{5j}^{m+1} := \frac{1}{2}\mathbf{q}_{2j}^m + \frac{1}{2}\mathbf{q}_{3j}^m \tag{3–8}$$

where

$$\mathbb{b}_j := \frac{1}{n}\left(\hat{\mathbb{b}}_0 + c_{j:n} + \frac{1}{2}c_{2j:n} + \frac{1}{8}c_{3j:n}\right), \tag{3–9}$$

$$\hat{\mathbb{b}}_0 := \frac{1}{2}, \qquad \mathbb{a} := \hat{\mathbb{b}}_0 - \frac{1}{4}, \qquad \mathbb{c} := \frac{11}{12}, \qquad \mathbb{d}_j := -\frac{1}{6n}c_{j:n}. \tag{3–10}$$

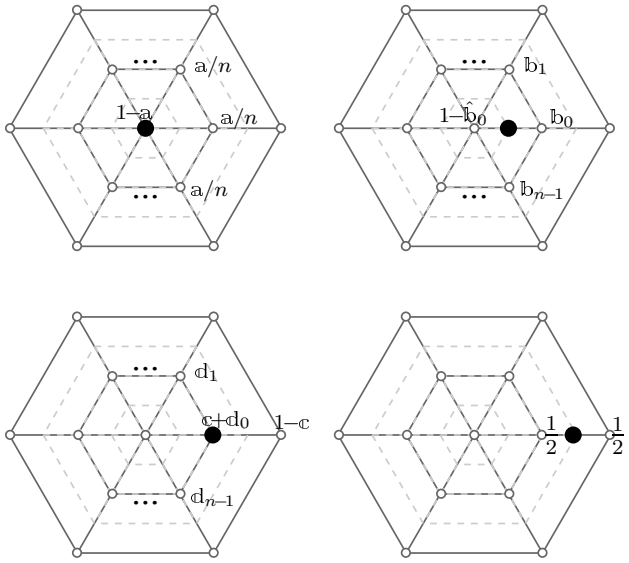


Figure 3-3. Radial Taylor subdivision rules. These masks show how to compute each vertex (•) on the refined mesh (*light gray, dashed*) based on the old mesh (*dark gray*, vertices as ∘). The special radial subdivision rules for RTS are isolated to the triangle fan at the center of the polar configuration. Further out, standard cubic rules are applied.

The limit surface is parameterized by $\mathbf{x}^{\mathrm{RTS}}$*, defined by the spline rings* $G_m\mathbf{q}^m$ *defined in* (3–3).

The subdivision rules of RTS are illustrated in Figure 3-3. It follows by definition that $n_m = n$. Since $\mathbf{q}_{3j}^{m+1}$, $\mathbf{q}_{4j}^{m+1}$, $\mathbf{q}_{5j}^{m+1}$ are computed via uniform cubic spline subdivision, $G_m\mathbf{q}^m$ defines a subset of $\mathbf{x}^{\mathrm{RTS}}$. The refinement weights for bi-3 polar subdivision are identical to those of RTS, except that it uses uniform cubic subdivision for $\mathbf{q}_{2j}^{m+1}$. As was the case for bi-3 polar subdivision [Karčiauskas and Peters, 2007a], we assume, that the polar valence $n \geq 5$. This assumption is not applicable to the other two subdivision algorithms, RTS_∞ and C^2PS, that we define later.

RTS can more compactly be written in terms of matrix multiplication:

$$\mathbf{q}^{m+1} = A\mathbf{q}^m, \tag{3–11}$$

where A is a $6n \times 6n$ matrix. Since the subdivision algorithm is rotationally symmetric around the polar vertex, our enumeration of the control points in $\mathbf{q}^m$ allows us to write A

in block circulant form, distributing evenly the contribution of the polar vertex amongst its n different labels $\mathbf{q}^m_{0j}$, $j \in \mathbb{Z}_n$.

$$A := \begin{bmatrix} A_0 & A_1 & \cdots & A_{n-1} \\ A_{n-1} & A_0 & \cdots & A_{n-2} \\ \vdots & & \ddots & \vdots \\ A_1 & \cdots & A_{n-1} & A_0 \end{bmatrix}$$

where

$$A_0 := \begin{bmatrix} (1-\mathbb{a})/n & \mathbb{a}/n & 0 & 0 & 0\,0 \\ (1-\hat{\mathbb{b}}_0)/n & \mathbb{b}_0 & 0 & 0 & 0\,0 \\ 0 & \mathbb{c}+\mathbb{d}_0 & 1-\mathbb{c} & 0 & 0\,0 \\ 0 & 1/2 & 1/2 & 0 & 0\,0 \\ 0 & 1/8 & 3/4 & 1/8 & 0\,0 \\ 0 & 0 & 1/2 & 1/2 & 0\,0 \end{bmatrix}, \qquad A_j := \begin{bmatrix} (1-\mathbb{a})/n & \mathbb{a}/n & 0\,0\,0\,0 \\ (1-\hat{\mathbb{b}}_0)/n & \mathbb{b}_j & 0\,0\,0\,0 \\ 0 & \mathbb{d}_j & 0\,0\,0\,0 \\ 0 & 0 & 0\,0\,0\,0 \\ 0 & 0 & 0\,0\,0\,0 \\ 0 & 0 & 0\,0\,0\,0 \end{bmatrix}, \text{ for } j \neq 0.$$

Only the first 3×3 block of the subdivision matrix has non-standard weights, while the rest is merely the application of uniform cubic spline subdivision in the radial direction. Nevertheless, the entire 6×6 matrix is required to define the spline rings for analysis.

3.3 Analysis

Using tools summarized in [Peters and Reif, 2008], we analyze RTS the subdivision limit by examining the limit

$$\lim_{m\to\infty} G_m(\mathbf{q}^m) = \lim_{m\to\infty} G_m(A^m\mathbf{q}) \tag{3–12}$$

of the sequence of spline rings defining $\mathbf{x}^{\text{RTS}}$ near the *pole*. Section 3.3.1 examines the spectrum of A that motivates the choice of its entries. Section 3.3.2 then reformulates RTS in eigenspace to derive, in Section 3.3.3, an expansion of the dominant terms at the pole and conclude that in the limit $n \to \infty$, the limit surface is C^2 at the pole.

3.3.1 Spectral analysis of RTS

The subdivision algorithm (not necessarily the surface) is rotationally symmetric and periodic such that $\mathbf{q}^m_{i,j+n} = \mathbf{q}^m_{ij}$. This suggests that a Fourier transform in the circular direction may factor out the radial and circular behavior of the subdivision algorithm to aid our analysis. In other words, since the subdivision matrix A is block circulant

due to the rotational symmetry of RTS, we can diagonalize it using the discrete Fourier transform. The complex Fourier block matrix

$$\mathcal{F} := (\omega_n^{-jk}\mathcal{I})_{j,k\in\mathbb{Z}_n} = \begin{bmatrix} \mathcal{I} & \mathcal{I} & \mathcal{I} & \cdots & \mathcal{I} \\ \mathcal{I} & \omega_n^{-1}\mathcal{I} & \omega_n^{-2}\mathcal{I} & \cdots & \omega_n^{1}\mathcal{I} \\ \mathcal{I} & \omega_n^{-2}\mathcal{I} & \omega_n^{-4}\mathcal{I} & \cdots & \omega_n^{2}\mathcal{I} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{I} & \omega_n^{1}\mathcal{I} & \omega_n^{2}\mathcal{I} & \cdots & \omega_n^{-1}\mathcal{I} \end{bmatrix} \tag{3–13}$$

where $\omega := \exp\left(\frac{2\pi\sqrt{-1}}{n}\right)$ and $\mathcal{I}$ is the 6×6 identity matrix. It can easily be verified that $\mathcal{F}$ is almost orthogonal:

$$\mathcal{F}^{-1} = \frac{1}{n}\mathcal{F}^* = \frac{1}{n}(\omega_n^{jk}\mathcal{I})_{j,k\in\mathbb{Z}_n},$$

where $\mathcal{F}^*$ is the Hermitian adjoint (conjugate transpose) of $\mathcal{F}$. An important property that is exploited later is that for $k > 0$, the k^{th} and $n-k^{\text{th}}$ block columns of $\mathcal{F}^{-1}$ are complex conjugates of each other. We can now block diagonalize A via

$$\hat{A} := \mathcal{F}A\mathcal{F}^{-1} = \begin{bmatrix} \hat{A}_0 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \hat{A}_1 & & \mathbf{0} \\ \vdots & & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \hat{A}_{n-1} \end{bmatrix}, \qquad \hat{A}_k := \sum_j^n \omega^{-jk} A_j,$$

where

$$\hat{A}_0 = \begin{bmatrix} 1-\mathtt{a} & \mathtt{a} & 0 & 0 & 0 & 0 \\ 1-\hat{\mathbb{b}}_0 & \hat{\mathbb{b}}_0 & 0 & 0 & 0 & 0 \\ 0 & \mathtt{c}+\hat{\mathtt{d}}_0 & 1-\mathtt{c} & 0 & 0 & 0 \\ 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 1/8 & 3/4 & 1/8 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 \end{bmatrix}, \qquad \hat{A}_k = \hat{A}_{n-k} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \hat{\mathbb{b}}_k & 0 & 0 & 0 & 0 \\ 0 & \mathtt{c}+\hat{\mathtt{d}}_k & 1-\mathtt{c} & 0 & 0 & 0 \\ 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 1/8 & 3/4 & 1/8 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 \end{bmatrix}$$

$$\hat{\mathbb{b}}_k := \sum_j^n \omega^{-jk}\mathbb{b}_j, \qquad \hat{\mathtt{d}}_k := \sum_j^n \omega^{-jk}\mathtt{d}_j.$$

Note that this re-definition of $\hat{\mathbb{b}}_0$ is consistent with its usage in the definition (3–9) of $\mathbb{b}_j$. Consequently,

$$\hat{\mathbb{b}}_k = \begin{cases} \frac{1}{2} & \text{if } k = 0 \\ \frac{1}{2} & \text{if } k \in \{1, n-1\} \\ \frac{1}{4} & \text{if } k \in \{2, n-2\} \\ \frac{1}{16} & \text{if } k \in \{3, n-3\} \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad \hat{\mathbb{d}}_k = \begin{cases} -\frac{1}{12} & \text{if } k \in \{1, n-1\} \\ 0 & \text{otherwise} \end{cases}$$

$\hat{A}_k$ is called the k^{th} *Fourier block* of $\hat{A}$ and encodes the action of RTS on the k^{th} frequency mode when going around the polar vertex. For instance, $\hat{A}_0$ alone determines the effect of RTS when each control point $\mathbf{q}_{ij}$ is independent of the circular index j. This includes polar configurations sampled from a constant function or a parabola.

Since the eigenvalues and eigenvectors of A and $\hat{A}$ are closely related by the Fourier transform, we use a similar notation for spectral analysis.

- $\ell_0, \ell_1, \ldots, \ell_{6n-1}$ are the eigenvalues of the subdivision matrix A (and also $\hat{A}$) in non-increasing order of magnitude: $|\ell_0| \geq |\ell_1| \geq \ldots \geq |\ell_{6n-1}|$. Equal eigenvalues are listed once for each multiplicity and treated separately.
- α_k is the index of Fourier block $\hat{A}_{\alpha_k}$ contributing eigenvalue ℓ_k, chosen so that $\ell_{k_1} = \ell_{k_2}$ and $k_1 < k_2$ imply $\alpha_{k_1} < \alpha_{k_2}$.
- $\mathbf{v}_k$ (respectively, $\mathbf{w}_k$) is the $6n$-dimensional right (respectively, left) real eigenvector of A corresponding to eigenvalue ℓ_k.
- $\hat{\mathbf{v}}_k$ (respectively, $\hat{\mathbf{w}}_k$) is the 6-dimensional right (respectively, left) eigenvector of the Fourier block $\hat{A}_{\alpha_k}$, corresponding to eigenvalue ℓ_k. $\hat{\mathbf{v}}_k$ is also called a *radial eigenvector*.

We abuse the caret ($\hat{}$) notation to represent objects or functions in the Fourier domain, even if they do not directly arrive via a Fourier transform. For example, as seen shortly in (3–14), the operator $\hat{G}_m$ (see (3–4)) defines the radial limit curve of $\hat{\mathbf{v}}_k$ in a manner similar to G_m defining the limit surface $\mathbf{x}$.

As summarized in Table 3-1, we have

$$(\ell_0, \ell_1, \ell_2, \ell_3, \ell_4, \ell_5) = (1, \lambda, \lambda, \mu, \mu, \mu)$$

Table 3-1. The dominant spectral behavior of $\hat{A}$. The left ($\hat{\mathbf{w}}_k$) and right ($\hat{\mathbf{v}}_k$) eigenvectors are normalized so that the related vectors $\mathbf{w}_k$ and $\mathbf{v}_k$ satisfy $\mathbf{w}_{k_1}^{\mathrm{T}}\mathbf{v}_{k_2} = \delta_{k_1 k_2}$.

	Fourier	eigen-		
k	block (α_k)	value (ℓ_k)	vector (right) ($\hat{\mathbf{v}}_k^{\mathrm{T}}$)	vector (left) ($\hat{\mathbf{w}}_k^{\mathrm{T}}$)
0	0	1	$[1,1,1,1,1,1]$	$\frac{1}{3}[2,1,0,0,0,0]$
1	1	1/2	$[0,1,2,3,4,5]$	$[0,2,0,0,0,0]$
2	$n-1$	1/2	$[0,1,2,3,4,5]$	$[0,2,0,0,0,0]$
3	0	1/4	$\frac{1}{3}[-1,2,11,26,47,74]$	$[-1,1,0,0,0,0]$
4	2	1/4	$\frac{1}{3}[0,2,11,26,47,74]$	$[0,3,0,0,0,0]$
5	$n-2$	1/4	$\frac{1}{3}[0,2,11,26,47,74]$	$[0,3,0,0,0,0]$

with $\lambda := \frac{1}{2}$ and $\mu := \lambda^2 = \frac{1}{4}$. The rest of the eigenvalues are real and positive with magnitude strictly less than ℓ_5. The eigenvalues $\ell_0 = 1$ and $\ell_3 = \mu$ are from $\hat{A}_0$; $\ell_1 = \ell_2 = \lambda$ are from $\hat{A}_1$ and $\hat{A}_{n-1}$; and the final two $\ell_4 = \ell_5 = \mu$ are from $\hat{A}_2$ and $\hat{A}_{n-2}$. In order for these five important Fourier blocks to exist, the valence n must be at least 5, justifying this assumption.

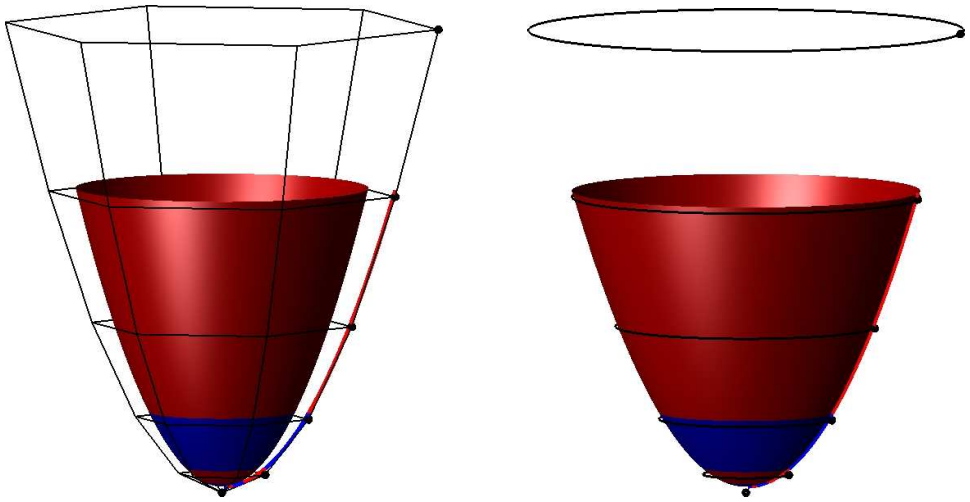
Since A is an operation on polar configurations, the eigenvectors $\mathbf{v}_k$ of A are also polar configurations. The *eigenspline* $e_k^{\mathrm{RTS}} : [0,4] \times \mathbb{R}_1 \to \mathbb{R}$ is the limit surface $\mathcal{L}(\mathbf{v}_k)$ of these polar configurations. The *radial eigenspline* $\hat{e}_k : [0,4] \to \mathbb{R}$ is the limit curve of radial eigenvector $\hat{\mathbf{v}}_k$ as defined by the decomposition

$$\hat{e}_k(r)\big|_{r\in[2\lambda^m,4\lambda^m]} := \left(\hat{G}_m\left(\hat{A}_{\alpha_k}^m \hat{\mathbf{v}}_k\right)\right)(r). \tag{3–14}$$

Figure 3-4 illustrates the relationship, for example, between $\mathbf{v}_3$, $\hat{\mathbf{v}}_3$, e_3^{RTS}, and $\hat{e}_3$.

Let $\Lambda := \mathrm{diag}(\ell_0, \ell_1, \ldots, \ell_{6n-1})$ be a diagonal matrix of the eigenvalues of A (respectively $\hat{A}$), and $\hat{V}$ be a matrix whose columns enumerate the corresponding right eigenvectors (of any scale) of $\hat{A}$, so that $\hat{A}\hat{V} = \hat{V}\Lambda$. Then,

$$\begin{aligned} & \hat{A}\hat{V} = \hat{V}\Lambda \\ \Rightarrow\ & \mathcal{F}A\mathcal{F}^{-1}\hat{V} = \hat{V}\Lambda \quad \Rightarrow \quad A(\mathcal{F}^{-1}\hat{V}) = \underbrace{(\mathcal{F}^{-1}\hat{V})}_{V_{\mathbb{C}}:=}\Lambda \\ \Rightarrow\ & AV_{\mathbb{C}} = V_{\mathbb{C}}\Lambda, \end{aligned} \tag{3–15}$$



A. $n = 6$ B. $n \to \infty$

Figure 3-4. Eigenvector $\mathbf{v}_3$ is the n-valent polar configuration (*black mesh/curves*) defining the eigenspline e_3^{RTS} (*blue and red surface*). The corresponding radial eigenvector $\hat{\mathbf{v}}_3$ (*indicated by* $\bullet$) defines the radial eigenspline $\hat{e}_3$ (*blue and red curve*). When $n \to \infty$, then $\hat{e}_3(r) = e_3^{\mathrm{RTS}}(r, 0)$.

implying that the columns of $V_{\mathbb{C}}$ are (complex) eigenvectors of A. Since the Fourier blocks $\hat{A}_k$ and $\hat{A}_{n-k}$ are identical and the corresponding pairs of block columns of $\mathcal{F}^{-1}$ are complex conjugate, eigenvalues from these Fourier blocks are associated via $V_{\mathbb{C}} = \mathcal{F}^{-1}\hat{V}$ with pairs of complex conjugate eigenvectors of A. The real eigenvectors $\mathbf{v}_k$ of A can hence be computed as the real and imaginary portions of these complex eigenvectors:

$$(\mathbf{v}_k)_{ij} := (\hat{\mathbf{v}}_k)_i \;\; \mathrm{op}_k\left(\frac{j}{n}\right), \qquad \mathrm{op}_k(\eta) := \begin{cases} c_{\alpha_k \eta} & \text{if } \alpha_k \leq n/2 \\ -s_{\alpha_k \eta} & \text{otherwise} \end{cases} \tag{3–16}$$

where $k \in \mathbb{Z}_{6n}$, $i \in \mathbb{Z}_6$, and $j \in \mathbb{Z}_n$. According to Table 3-1, $(\mathrm{op}_0(\eta), \mathrm{op}_1(\eta), \ldots, \mathrm{op}_5(\eta)) = (1, c_\eta, s_\eta, 1, c_{2\eta}, s_{2\eta})$. The eigenspline e_k^{RTS} inherits the tensored nature of $\mathbf{v}_k$.

Lemma 1. *For any* $f : \mathbb{R} \to \mathbb{R}$, *the operator* $\mathrm{B}^n : f \mapsto \sum_j^n f\left(\frac{j}{n}\right) \mathring{N}_j^{(n)}$ *uses uniform samples on a function* f *as the control points of a periodic spline. Then, the eigenspline* e_k^{RTS} *decomposes according to*

$$e_k^{\mathrm{RTS}}(r, \tau) = \hat{e}_k(r) \;\; \mathrm{B}^n \mathrm{op}_k(\tau) \tag{3–17}$$

Proof. For $r \in [2\lambda^m, 4\lambda^m]$,

$$
\begin{aligned}
e_k^{\mathrm{RTS}}(r,\tau) &\overset{(3\text{–}3)}{=} \left(G_m \mathbf{v}^m\right)(r,\tau) \\
&\overset{\substack{(3\text{–}1)\\(3\text{–}11)}}{=} \sum_{i=1}^{5}\sum_{j}^{n} \left(A^m \mathbf{v}\right)_{ij} \overset{*}{N}{}_i^{(m)}(r) \overset{\circ}{N}{}_j^{(n)}(\tau) \\
&= \sum_{i=1}^{5}\sum_{j}^{n} \ell_k^m \mathbf{v}_{ij} \overset{*}{N}{}_i^{(m)}(r) \overset{\circ}{N}{}_j^{(n)}(\tau) \\
&\overset{(3\text{–}16)}{=} \sum_{i=1}^{5}\sum_{j}^{n} \ell_k^m (\hat{\mathbf{v}}_k)_i \;\; \mathrm{op}_k\left(\frac{j}{n}\right) \;\; \overset{*}{N}{}_i^{(m)}(r) \overset{\circ}{N}{}_j^{(n)}(\tau) \\
&= \sum_{i=1}^{5}\sum_{j}^{n} \left(\hat{A}_{\alpha_k}^m(\hat{\mathbf{v}}_k)\right)_i \;\; \mathrm{op}_k\left(\frac{j}{n}\right) \;\; \overset{*}{N}{}_i^{(m)}(r) \overset{\circ}{N}{}_j^{(n)}(\tau) \\
&= \left(\sum_{i=1}^{5} \left(\hat{A}_{\alpha_k}^m(\hat{\mathbf{v}}_k)\right)_i \overset{*}{N}{}_i^{(m)}(r)\right) \left(\sum_{j}^{n} \mathrm{op}_k\left(\frac{j}{n}\right) \overset{\circ}{N}{}_j^{(n)}(\tau)\right) \\
&\overset{(3\text{–}4)}{=} \left(\hat{G}_m\left(\hat{A}_{\alpha_k}^m(\hat{\mathbf{v}}_k)\right)\right)(r) \;\; \mathrm{B}^n \mathrm{op}_k(\tau) \\
&\overset{(3\text{–}14)}{=} \hat{e}_k(r) \;\; \mathrm{B}^n \mathrm{op}_k(\tau)
\end{aligned}
$$

□

Lemma 1 shows that the block diagonalization factors the radial from the circular. From this formulation, it is also obvious that e_k^{RTS} is periodic in τ with a period of $\frac{1}{\alpha_k}$, which is a direct consequence of $\hat{\mathbf{v}}_k$ having frequency mode α_k. Eigensplines and radial eigensplines also inherit the scaling property of eigenvectors, in that for $\bar{r} = \lambda r$,

$$
\begin{aligned}
\hat{e}_k(\lambda r)\big|_{r\in[2\lambda^m,4\lambda^m]} &= \hat{e}_k(\bar{r})\big|_{\bar{r}\in[2\lambda^{m+1},4\lambda^{m+1}]} \\
\overset{(3\text{–}14)}{=} \left(\hat{G}_{m+1}\left(\hat{A}_{\alpha_k}^{m+1}\hat{\mathbf{v}}\right)\right)(\bar{r}) &= \ell_k \left(\hat{G}_{m+1}\left(\hat{A}_{\alpha_k}^{m}\hat{\mathbf{v}}\right)\right)(\bar{r}) \\
= \ell_k \left(\hat{G}_m\left(\hat{A}_{\alpha_k}^{m}\hat{\mathbf{v}}\right)\right)(r) &\overset{(3\text{–}14)}{=} \ell_k \hat{e}_k(r)\big|_{r\in[2\lambda^m,4\lambda^m]},
\end{aligned}
\tag{3–18}
$$

implying that $\hat{e}_k(\lambda r) = \ell_k \hat{e}_k(r)$, and, due to Lemma 1, that $e_k^{\mathrm{RTS}}(\lambda r, \tau) = \ell_k e_k^{\mathrm{RTS}}(r,\tau)$.

With this scaling relationship, the first six radial eigensplines can be written out explicitly. The subdivision matrix A was constructed to have the spectral behavior in

Table 3-1 in order to satisfy the following key lemma. For succinctness and clarity, we use the notation $\mathrm{B}^n c_{\alpha\tau}$ for $\left(\mathrm{B}^n(\gamma \mapsto c_{\alpha\gamma})\right)(\tau)$, where the operator B^n is applied to the function $\gamma \mapsto c_{\alpha\gamma} := \cos(2\pi\alpha\gamma)$, and the resulting spline is evaluated at τ. Similarly, $\left(\mathrm{B}^n(\gamma \mapsto s_{\alpha\gamma})\right)(\tau)$ is shortened to $\mathrm{B}^n s_{\alpha\tau}$.

Lemma 2 (Reproduction of Radial Taylor Basis Functions)**.** *For $r \in [0,4]$,*

$$\hat{e}_0(r) = 1 \tag{3–19}$$

$$\hat{e}_1(r) = \hat{e}_2(r) = r \tag{3–20}$$

$$\hat{e}_3(r) = \hat{e}_4(r) = \hat{e}_5(r) = r^2 \tag{3–21}$$

$$\hat{e}_k(r) = \mathrm{o}(r^2) \quad \textit{as } r \to 0 \textit{ for } k > 5 \tag{3–22}$$

which imply, by Lemma 1, that

$$\begin{aligned}
&e_0^{\mathrm{RTS}}(r,\tau) = 1, \quad && e_1^{\mathrm{RTS}}(r,\tau) = r\ \mathrm{B}^n c_\tau, \quad && e_2^{\mathrm{RTS}}(r,\tau) = r\ \mathrm{B}^n s_\tau,\\
&e_3^{\mathrm{RTS}}(r,\tau) = r^2, \quad && e_4^{\mathrm{RTS}}(r,\tau) = r^2\ \mathrm{B}^n c_{2\tau}, \quad && e_5^{\mathrm{RTS}}(r,\tau) = r^2\ \mathrm{B}^n s_{2\tau}\\
&e_k^{\mathrm{RTS}}(r,\tau) = \mathrm{o}(r^2) \quad \textit{as } r \to 0 \textit{ for } k > 5.
\end{aligned}$$

Proof. (3–19) follows since $\hat{A}_0\hat{\mathbf{v}}_0 = \hat{\mathbf{v}}_0$ and $\left(\hat{G}_0\hat{\mathbf{v}}_0\right)(r) = 1$. We can now verify, for $k \in \{1,2\}$, that $\hat{e}_k(r)|_{r\in[2,4]} = \left(\hat{G}_0\hat{\mathbf{v}}_k\right)(r) = r|_{r\in[2,4]}$ by B-spline-to-power-series conversion (2–1). The additional property from (3–18) that $\hat{e}_k(\frac{1}{2}r) = \frac{1}{2}\hat{e}_k(r)$ implies (3–20). Similarly, for $k \in \{3,4,5\}$, B-spline-to-power-series conversion shows that $\hat{e}_k(r)|_{r\in[2,4]} = r^2|_{r\in[2,4]}$. Hence, $\hat{e}_k(\frac{1}{2}r) = \frac{1}{4}\hat{e}_k(r)$ implies (3–21).

When $k > 5$, using $\bar{r} := \frac{r}{\lambda^m}$,

$$\begin{aligned}
\hat{e}_k(r)\Big|_{r\in[2\lambda^m,4\lambda^m]} &= \hat{e}_k(\lambda^m\bar{r})\Big|_{\bar{r}\in[2,4]} \overset{(3\text{–}18)}{=} \underbrace{\ell_k^m}_{\mathrm{o}(\mu^m)}\hat{e}_k(\bar{r})\Big|_{\bar{r}\in[2,4]}\\
= \mathrm{o}(\mu^m)\Big|_{\bar{r}\in[2,4]} &= \mathrm{o}\left((\lambda^m\bar{r})^2\right)\Big|_{\bar{r}\in[2,4]} = \mathrm{o}(r^2)\Big|_{r\in[2\lambda^m,4\lambda^m]},
\end{aligned}$$

proving (3–22). ∎

Lemma 2 is used to derive a quadratic Taylor expansion to show second-order continuity for two of our proposed subdivision algorithms.

3.3.2 Reformulating RTS in eigenspace

Defining the real matrix $V_{\mathbb{R}}$ as having k^{th} column $\mathbf{v}_k$, we have (as in (3–15)),

$$AV_{\mathbb{R}} = V_{\mathbb{R}}\Lambda \quad \Rightarrow \quad V_{\mathbb{R}}^{-1}AV_{\mathbb{R}} = \Lambda \quad \Rightarrow \quad V_{\mathbb{R}}^{-1}A = \Lambda V_{\mathbb{R}}^{-1}, \tag{3–23}$$

implying that the rows of $V_{\mathbb{R}}^{-1}$ are left eigenvectors $\mathbf{w}_k$ of A. We choose normalization so that $\mathbf{w}_{k_1}^{\mathrm{T}}\mathbf{v}_{k_2} = \delta_{k_1k_2}$. Multiplication with $V_{\mathbb{R}}^{-1}$ projects the polar configuration into eigenspace using these left eigenvectors $\mathbf{w}_k$. Precisely, $\mathbf{q}$ (respectively, $\mathbf{x}^{\mathrm{RTS}}$) can be written as a linear combination of the $6n$ right eigenvectors $\mathbf{v}_k$ (respectively, eigensplines $\hat{e}$) as follows

$$\mathbf{q} = \underbrace{V_{\mathbb{R}}}_{\text{columns } \mathbf{v}_k}\ (\ \underbrace{V_{\mathbb{R}}^{-1}}_{\text{rows } \mathbf{w}_k}\ \mathbf{q}) \qquad \Rightarrow \quad \mathbf{q}_{ij} = \sum_k^{6n} \mathbf{p}_k\ \ (\mathbf{v}_k)_{ij} \tag{3–24}$$

$$\Rightarrow \mathcal{L}^{\mathrm{RTS}}(\mathbf{q}) = \sum_k^{6n} \mathbf{p}_k\ \ \mathcal{L}^{\mathrm{RTS}}(\mathbf{v}_k) \qquad \Rightarrow \quad \mathbf{x}^{\mathrm{RTS}}(r,\tau) = \sum_k^{6n} \mathbf{p}_k\ \ e_k^{\mathrm{RTS}}(r,\tau), \tag{3–25}$$

where each *eigencoefficient* $\mathbf{p}_k$ is the inner product of $\mathbf{w}_k$ and $\mathbf{q}$. As was the case for $\mathbf{v}_k$, $\mathbf{w}_k$ is computed from $\hat{\mathbf{w}}_k$ (listed in Table 3-1) using the inverse Fourier transform so that

$$\mathbf{p}_k := \mathbf{w}_k^{\mathrm{T}}\mathbf{q} = \sum_i^6 \sum_j^n \underbrace{\frac{1}{n}\ (\hat{\mathbf{w}}_k)_i\ \ \mathrm{op}_k\left(\frac{j}{n}\right)}_{(\mathbf{w}_k)_{ij}}\ \ \mathbf{q}_{ij} \tag{3–26}$$

Specifically, for $k \in \mathbb{Z}_6$,

$$\begin{aligned}
\mathbf{p}_0 &:= \frac{2}{3}\mathbf{q}_{00} + \frac{1}{3n}\sum_j^n \mathbf{q}_{1j}, & \mathbf{p}_3 &:= -\mathbf{q}_{00} + \frac{1}{n}\sum_j^n \mathbf{q}_{1j}, \\
\mathbf{p}_1 &:= \frac{2}{n}\sum_j^n c_{j:n}\mathbf{q}_{1j}, & \mathbf{p}_4 &:= \frac{3}{n}\sum_j^n c_{2j:n}\mathbf{q}_{1j}, \\
\mathbf{p}_2 &:= \frac{2}{n}\sum_j^n s_{j:n}\mathbf{q}_{1j}, & \mathbf{p}_5 &:= \frac{3}{n}\sum_j^n s_{2j:n}\mathbf{q}_{1j}
\end{aligned} \tag{3–27}$$

The eigencoefficient $\mathbf{p}_k^m := \mathbf{w}_k^{\mathrm{T}}\mathbf{q}^m$ of $\mathbf{q}^m$ simplifies to $\mathbf{p}_k^m = \mathbf{w}_k^{\mathrm{T}}A^m\mathbf{q} = (\mathbf{w}_k^{\mathrm{T}}A^m)\mathbf{q} = \ell_k^m\mathbf{w}_k^{\mathrm{T}}\mathbf{q} = \ell_k^m\mathbf{p}_k$. The refinement equations (3–5)–(3–8) of RTS can now be rewritten in terms of $\mathbf{p}_k$.

Lemma 3. *The m-times refined mesh $\mathbf{q}^m$ is defined by eigencoefficients and powers of eigenvalues as follows.*

$$\mathbf{q}_{00}^{m+1} = \mathbf{p}_0 - \frac{\mu^{m+1}}{3}\mathbf{p}_3 \tag{3–28}$$

$$\mathbf{q}_{1j}^{m+1} = \mathbf{p}_0 + \lambda^{m+1}(\mathbf{p}_1 c_{j:n} + \mathbf{p}_2 s_{j:n}) + \frac{2\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2j:n} + \mathbf{p}_5 s_{2j:n}) \tag{3–29}$$

$$\mathbf{q}_{2j}^{m+1} = \mathbf{p}_0 + 2\lambda^{m+1}(\mathbf{p}_1 c_{j:n} + \mathbf{p}_2 s_{j:n}) + \frac{11\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2j:n} + \mathbf{p}_5 s_{2j:n}) \tag{3–30}$$

$$\mathbf{q}_{3j}^{m+1} = \mathbf{p}_0 + 3\lambda^{m+1}(\mathbf{p}_1 c_{j:n} + \mathbf{p}_2 s_{j:n}) + \frac{26\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2j:n} + \mathbf{p}_5 s_{2j:n}) \tag{3–31}$$

$$\mathbf{q}_{4j}^{m+1} = \mathbf{p}_0 + 4\lambda^{m+1}(\mathbf{p}_1 c_{j:n} + \mathbf{p}_2 s_{j:n}) + \frac{47\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2j:n} + \mathbf{p}_5 s_{2j:n}) \tag{3–32}$$

$$\mathbf{q}_{5j}^{m+1} = \mathbf{p}_0 + 5\lambda^{m+1}(\mathbf{p}_1 c_{j:n} + \mathbf{p}_2 s_{j:n}) + \frac{74\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2j:n} + \mathbf{p}_5 s_{2j:n}) \tag{3–33}$$

Proof.

$$\begin{aligned}
\mathbf{q}_{00}^{m+1} &= \frac{3}{4}\mathbf{q}_{00}^m + \frac{1}{4n_m}\sum_h^{\mathcal{G}\mathbf{q}^m}\mathbf{q}_{1,[h]}^m \overset{(3\text{–}27)}{=} \mathbf{p}_0^m - \frac{1}{12}\mathbf{p}_3^m \\
&= \mathbf{p}_0 - \frac{\mu^m}{12}\mathbf{p}_3 = \mathbf{p}_0 - \frac{\mu^{m+1}}{3}\mathbf{p}_3 \\
\mathbf{q}_{1j}^{m+1} &= \frac{1}{2}\mathbf{q}_{00}^m + \frac{1}{n}\sum_h^n\left(\frac{1}{2} + c_{h-j:n} + \frac{1}{2}c_{2(h-j):n}\right)\mathbf{q}_{1h}^m \\
&\overset{\substack{\text{addition rule}\\ \text{for cosine \& (3–27)}}}{=} \mathbf{p}_0^m + \frac{1}{2}(\mathbf{p}_1^m c_{j:n} + \mathbf{p}_2^m s_{j:n}) + \frac{1}{6}(\mathbf{p}_3^m + \mathbf{p}_4^m c_{2j:n} + \mathbf{p}_5^m s_{2j:n}) \\
&= \mathbf{p}_0 + \lambda^{m+1}(\mathbf{p}_1 c_{j:n} + \mathbf{p}_2 s_{j:n}) + \frac{2\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2j:n} + \mathbf{p}_5 s_{2j:n})
\end{aligned}$$

The proof for (3–30)–(3–33) is similar to that of (3–29). □

With the radial eigenvectors $\hat{\mathbf{v}}_0 = [1,1,1,1,1,1]$, $\hat{\mathbf{v}}_1 = \hat{\mathbf{v}}_2 = [0,1,2,3,4,5]$, $\hat{\mathbf{v}}_3 = \frac{1}{3}[0,2,11,26,47,74]$, and $\hat{\mathbf{v}}_4 = \hat{\mathbf{v}}_5 = \frac{1}{3}[0,2,11,26,47,74]$, (3–28)–(3–33) can be written as

$$
\begin{aligned}
\mathbf{q}_{ij}^{m+1} &= \mathbf{p}_0 + (\hat{\mathbf{v}}_1)_i \lambda^{m+1} \left(\mathbf{p}_1 c_{j:n} + \mathbf{p}_2 s_{j:n}\right) \\
&\quad + \mu^{m+1} \left((\hat{\mathbf{v}}_3)_i \mathbf{p}_3 + (\hat{\mathbf{v}}_4)_i \left(\mathbf{p}_4 c_{2j:n} + \mathbf{p}_5 s_{2j:n}\right)\right) \\
\Rightarrow \mathbf{q}_{i,[\tau]}^{m+1} &= \mathbf{p}_0 + (\hat{\mathbf{v}}_1)_i \lambda^{m+1} \left(\mathbf{p}_1 c_{\tau} + \mathbf{p}_2 s_{\tau}\right) \\
&\quad + \mu^{m+1} \left((\hat{\mathbf{v}}_3)_i \mathbf{p}_3 + (\hat{\mathbf{v}}_4)_i \left(\mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}\right)\right) \qquad (3\text{–}34)
\end{aligned}
$$

for $i \in \mathbb{Z}_6$. In particular,

$$
\lim_{m\to\infty} \mathbf{q}_{ij}^{m} = \mathbf{p}_0 \quad \Rightarrow \quad \lim_{m\to\infty} \left(G_m \mathbf{q}^m\right)(r,\tau) = \mathbf{p}_0,
$$

converging to a unique point $\mathbf{p}_0$, showing that $\mathbf{x}^{\mathrm{RTS}}(r,\tau)$ is C^0 at the pole.

3.3.3 Eigenspace expansion and curvature continuity

C^0 and C^1 continuity can be seen more explicitly by expressing $\mathbf{x}^{\mathrm{RTS}}$ using the eigenprojection.

$$
\begin{aligned}
\mathbf{x}^{\mathrm{RTS}}(r,\tau) &\overset{(3\text{–}25)}{=} \sum_{k}^{6n_0} \mathbf{p}_k \, e_k^{\mathrm{RTS}}(r,\tau) \\
&\overset{\text{Lemma 2}}{=} \mathbf{p}_0 \, e_0^{\mathrm{RTS}}(r,\tau) + \left(\mathbf{p}_1 \, e_1^{\mathrm{RTS}}(r,\tau) + \mathbf{p}_2 \, e_2^{\mathrm{RTS}}(r,\tau)\right) \\
&\quad + \left(\mathbf{p}_3 \, e_3^{\mathrm{RTS}}(r,\tau) + \mathbf{p}_4 \, e_4^{\mathrm{RTS}}(r,\tau) + \mathbf{p}_5 \, e_5^{\mathrm{RTS}}(r,\tau)\right) + \mathrm{o}\left(r^2\right) \qquad (3\text{–}35) \\
&\overset{\text{Lemma 2}}{=} \mathbf{p}_0 + \left(\mathbf{p}_1 r \, \mathrm{B}^n c_{\tau} + \mathbf{p}_2 r \, \mathrm{B}^n s_{\tau}\right) + \left(\mathbf{p}_3 r^2 + \mathbf{p}_4 r^2 \, \mathrm{B}^n c_{2\tau} + \mathbf{p}_5 r^2 \, \mathrm{B}^n s_{2\tau}\right) + \mathrm{o}\left(r^2\right)
\end{aligned}
$$

This expansion using the eigensplines is almost a Taylor expansion. [Peters and Reif, 2008, Section 5.2] shows that the *characteristic spline* $\chi(r,\tau) := (e_1^{\mathrm{RTS}}(r,\tau), e_2^{\mathrm{RTS}}(r,\tau)) = r(\mathrm{B}^n c_{\tau}, \mathrm{B}^n s_{\tau})$ is the only reparameterization, up to linear transformation, of $\mathbf{x}^{\mathrm{RTS}}$ that can reproduce a linear Taylor expansion at the pole. Figures 2-7B, and 3-2 illustrate the spline rings of χ, which are regular and injective, validating the reparameterization. Using $(x,y) := \chi(r,\tau)$, the reparameterized surface $\overline{\mathbf{x}}^{\mathrm{RTS}}(x,y) := \mathbf{x}^{\mathrm{RTS}}(r,\tau)$ has the first-order Taylor expansion

$$\overline{\mathbf{x}}^{\mathrm{RTS}}(x,y) = \mathbf{p}_0 + (\mathbf{p}_1 x + \mathbf{p}_2 y) + \mathrm{o}\,(r)\,\Big|_{r\in[2\lambda^m, 4\lambda^m]}$$

and is therefore C^1 at the pole. By the conditions in [Peters and Reif, 2008, Section 7.1], the additional property that $(\ell_0, \ell_1, \ell_2, \ell_3, \ell_4, \ell_5) = (1, \lambda, \lambda, \lambda^2, \lambda^2, \lambda^2)$ implies that the limit surface also has bounded curvature.

In order to take this a step further to C^2 continuity, we need eigensplines e_3^{RTS}, e_4^{RTS}, e_5^{RTS} to be quadratic with respect to the reparameterization χ to induce a second-order Taylor expansion. Preferably, these eigensplines should not be zero, since that would result in a zero second derivative and a visible flat-spot in the vicinity of the pole. $\chi(r,\tau) = r(\mathrm{B}^n c_\tau, \mathrm{B}^n s_\tau)$ is degree 1 in r and degree 3 in τ, which is the minimum degree needed to create C^2 spline rings around the pole. Consequently, e_3^{RTS}, e_4^{RTS}, and e_5^{RTS} need degree $(2,6)$ to be quadratic in χ. This implies that it is impossible to create a stationary C^2 subdivision for polar configurations based on uniform splines with degree less than 6 in the circular direction.

However, in the limit $n \to \infty$, the surface around the pole is no longer a spline in the circular direction, but an arbitrary curve $\mathbf{q}_{i,[\tau]}$ for $\tau \in \mathbb{R}_1$ (Figure 3-4B). Denote this case as RTS_∞, with $\mathbf{q}_{i,[\tau]}$ being the *control curves* of this subdivision algorithm. We now show that a non-trivial second-order Taylor expansion exists at the pole for RTS_∞.

Lemma 4. $e_k^{\mathrm{RTS}}(r,\tau) - \hat{e}_k(r)\,\mathrm{op}_k(\tau) = \mathrm{O}\left(\frac{1}{n^2}\right)$, *implying that*

$e_k^{\mathrm{RTS}_\infty}(r,\tau) := \lim_{n\to\infty} e_k^{\mathrm{RTS}}(r,\tau) = \hat{e}_k(r)\,\mathrm{op}_k(\tau)$

Proof. Since $\hat{e}_k(r)$ is independent of valence, and

$$\begin{aligned} e_k^{\mathrm{RTS}}(r,\tau) - \hat{e}_k(r)\,\mathrm{op}_k(\tau) &= \hat{e}_k(r)\,\mathrm{B}^n\mathrm{op}_k(\tau) - \hat{e}_k(r)\,\mathrm{op}_k(\tau) \\ &= \hat{e}_k(r)\,(\mathrm{B}^n\mathrm{op}_k(\tau) - \mathrm{op}_k(\tau))\,, \end{aligned}$$

we need only examine the spline approximation of $\mathrm{op}_k(\tau)$ via B^n. We show that

1. the distance between $\mathrm{B}^n\mathrm{op}_k(\tau)$ and its control polygon is $\mathrm{O}\left(\frac{1}{n^2}\right)$, and that

2. $\mathrm{op}_k(\tau)$ and its linear interpolant (i.e. the control polygon of $\mathrm{B}^n\mathrm{op}_k(\tau)$) is $\mathrm{O}\left(\frac{1}{n^2}\right)$.

Together, these statements imply by the triangle inequality that $e_k^{\mathrm{RTS}}(r,\tau) - \hat{e}_k(r)\,\mathrm{op}_k(\tau) = \mathrm{O}(\frac{1}{n^2})$, proving the lemma.

Step 1. [Lutterkort and Peters, 2001] showed that for uniform cubic splines with control points $[\mathbf{q}_{ij}]_{j\in\mathcal{Z}_n}$, the distance between the control polygon and the spline is proportional to the second differences of the control points: $\frac{1}{6}\max_j\{|\mathbf{q}_{i,j-1} - 2\mathbf{q}_{ij} + \mathbf{q}_{i,j+1}|\}$. In the context of this lemma, $\mathbf{q}_{ij} = \mathrm{op}_k\left(\frac{j}{n}\right)$, and

$$
\begin{aligned}
&\frac{1}{6}\max_j\{|\mathbf{q}_{i,j-1} - 2\mathbf{q}_{ij} + \mathbf{q}_{i,j+1}|\} \\
=&\frac{1}{6}\max_j\left\{\left|\underbrace{\mathrm{op}_k\left(\frac{j-1}{n}\right) + \mathrm{op}_k\left(\frac{j+1}{n}\right)}\; - 2\,\mathrm{op}_k\left(\frac{j}{n}\right)\right|\right\} \\
=&\frac{1}{6}\max_j\left\{\left|2\,\mathrm{op}_k\left(\frac{j}{n}\right)c_{\alpha_k:n} - 2\,\mathrm{op}_k\left(\frac{j}{n}\right)\right|\right\} \\
=&\frac{1}{3}\max_j\left\{\underbrace{\left|\mathrm{op}_k\left(\frac{j}{n}\right)\right|}_{\leq 1}|(c_{\alpha_k:n} - 1)|\right\} \\
\leq&\frac{1}{3}\max_j\{1 - c_{\alpha_k:n}\} \quad \underset{\text{expan.}}{\overset{\text{Taylor}}{=}} \quad \mathrm{O}\left(\frac{\alpha_k^2}{n^2}\right) \quad = \quad \mathrm{O}\left(\frac{1}{n^2}\right)
\end{aligned}
$$

Step 2. For an arbitrary function $f : [a,b] \in \mathbb{R} \to \mathbb{R}$, a Taylor expansion at a shows that a piecewise linear interpolant with distance $\frac{1}{n}$ between breakpoints approximates f with a deviation of $\mathrm{O}\left(\frac{1}{n^2}\max_{[a,b]}\{f''\}\right)$. Consequently, the piecewise linear interpolant to $\mathrm{op}_k(\tau)$ converges $\mathrm{O}\left(\frac{\alpha_k^2}{n^2}\right) = \mathrm{O}\left(\frac{1}{n^2}\right)$. □

Theorem 1. *In the limit $n \to \infty$, RTS is C^2 at the pole.*

Proof. Assuming $n \to \infty$ and continuing from (3–35),

$$
\begin{aligned}
&\mathbf{x}^{\mathrm{RTS}_\infty}(r,\tau)\Big|_{r\in[2\lambda^m,4\lambda^m]} \\
\underset{\text{Lemma 4}}{\overset{\text{Lemma 2 \&}}{=}}&\ \mathbf{p}_0 + (\mathbf{p}_1 r c_\tau + \mathbf{p}_2 r s_\tau) + \left(\mathbf{p}_3 r^2 + \mathbf{p}_4 r^2 c_{2\tau} + \mathbf{p}_5 r^2 s_{2\tau}\right) + \mathrm{o}\left(r^2\right)
\end{aligned}
$$

Changing from polar to Cartesian coordinates $(x, y) := (rc_\tau, rs_\tau)$, $\overline{\mathbf{x}}^{\mathrm{RTS}_\infty}(x, y) := \mathbf{x}^{\mathrm{RTS}_\infty}(r, \tau)$ reveals the following second-order Taylor expansion at the pole

$$\overline{\mathbf{x}}^{\mathrm{RTS}_\infty}(x, y) = \mathbf{p}_0 + (\mathbf{p}_1 x + \mathbf{p}_2 y) + \left(\mathbf{p}_3(x^2 + y^2) + \mathbf{p}_4(x^2 - y^2) + \mathbf{p}_5(2xy)\right) + \mathrm{o}\left(x^2 + y^2\right),$$

proving C^2 continuity at the pole. □

Nevertheless, curvature continuity comes at a cost: we are no longer polynomial in the circular direction. In the Chapter 4, we adapt the intuition developed so far to create a C^2 bi-3 subdivision algorithm that overcomes these disadvantages.

3.4 Approximation via Mesh Refinement

Mesh refinement is easiest to demonstrate on a control mesh with latitude-longitude connectivity of the earth as in Figures 2-5B and 2-6. Such a *spherical mesh* consists entirely of ordinary quads and exactly two polar configurations. Spherical meshes have precisely two directions: A) radial, or longitudinal, corresponding to the j-spokes of the polar configurations; and B) circular, or latitudinal, corresponding to the i-links of the polar configurations. Each radial sequence of control points of the spherical mesh is similarly called a *spoke*, while each (periodic) circular sequence is a *link*. We can perform *radial subdivision* along the spokes of a spherical mesh by using the special RTS rules of Definition 1 and Figure 3-3 in the vicinity of polar vertices, while using univariate cubic refinement (2–4) away from them. We can also double the valence of each polar vertex by performing *circular subdivision* along each link using univariate cubic refinement.

The RTS limit surface is defined by continually applying radial subdivision and interpreting links sufficiently far away from the polar vertex as the control points of a uniform bi-3 spline, implying that circular subdivision may be applied on these links. Consequently, the RTS limit surface of a spherical mesh can be computed a la Figure 2-2(A-B) by applying radial subdivision ad infinitum followed by circular subdivision ad infinitum. An m-times subdivided approximation this limit surface can hence be computed

by subdividing m times in the radial direction followed by m-times subdividing in the circular direction, as demonstrated in Figure 2-5B.

On the other hand, the curvature-continuous variation RTS_∞ requires each polar vertex to have infinite valence before radial subdivision is ever applied. This is accomplished by interpreting each link to be the control points of a cubic spline which acts as a control curve of RTS_∞. The corresponding limit surface can be computed by applying circular subdivision ad infinitum (converging to the control curves) followed by radial subdivision. As a result, an m-times refined approximation is computed by subdividing m times in the circular direction followed by m-times in the radial direction.

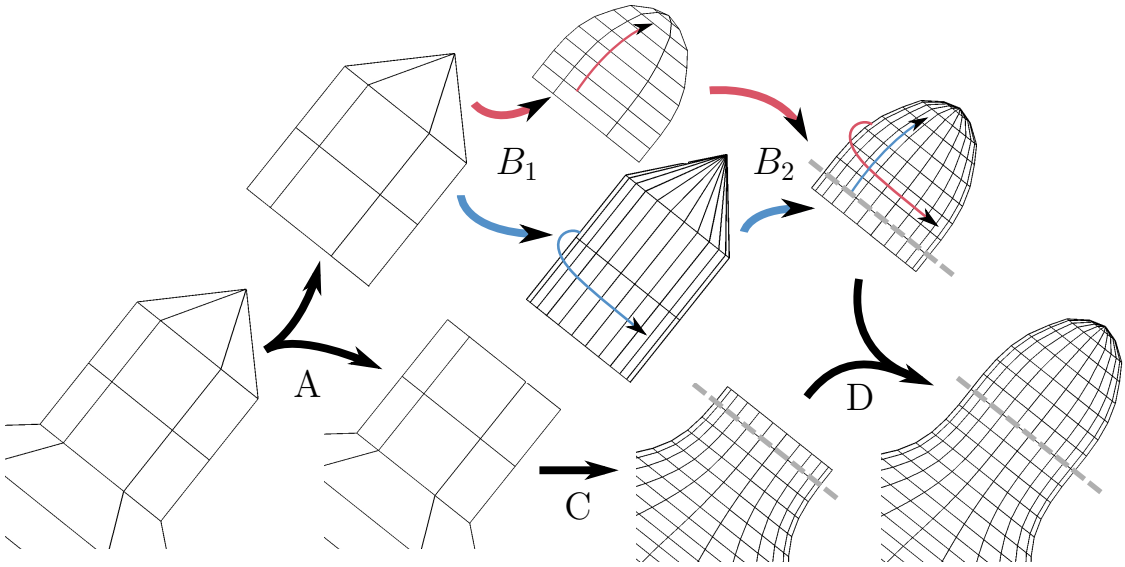


Figure 3-5. Combining Catmull-Clark and RTS. A) Separating the input mesh. B) Subdividing the polar configuration B_1) radially then B_2) circularly for bounded curvature (*red arrows*), *OR* B_1) circularly then B_2) radially for curvature continuity (*blue arrows*). C) Subdividing the remainder using Catmull-Clark. D) Joining the refined meshes after removal of overlapping facets.

Either mesh refinement technique can be combined with Catmull-Clark subdivision to be applicable to arbitrary quad meshes augmented with polar configurations (see Figure 3-5).

1. *Split off polar configurations:* Copy all polar 3-rings and remove each polar vertex from the input mesh.

2. *Subdivide polar configurations:* For each polar configuration,

   (a) subdivide m times radially, and then

   (b) subdivide m times in the circular direction.

3. *Subdivide the remaining mesh:* Apply m steps of Catmull-Clark subdivision to the mesh without the polar vertices.

4. *Merge results:* Drop the boundary facets of the meshes subdivided in steps 2 and 3 and join them by identifying the resulting geometrically identical boundary vertices.

Note that the 2- and 3-links are copied with the polar vertex, but not removed from the rest of the mesh (Figure 3-5A), and both Catmull-Clark and polar subdivision refine these common links using uniform bi-3 subdivision rules. The transition between the Catmull-Clark and polar limit surfaces is therefore C^2.

The disadvantage of such a refinement scheme is that it is not iterative. We cannot take the already-refined mesh and apply RTS radial subdivision to converge to the same limit surface. To avoid separation of the polar configuration from the rest of the surface, it would be far better if the subdivision algorithm offered a simultaneous radial/circular mesh refinement algorithm in the spirit of Catmull-Clark and Figure 2-3C. Chapter 4 describes such an algorithm.

CHAPTER 4
C^2 POLAR SUBDIVISION (C^2PS)

4.1 Semi-Stationary Subdivision

RTS can be adapted to non-stationary connectivity while keeping the weights dependent only on the connectivity of the mesh. In particular, the valence of the polar vertex doubles after every subdivision. We are no longer limited by stationary subdivision theory, which requires degree 6 in the circular direction for second-order continuity, as shown previously.

Definition 2. *Denote by* C^2 polar subdivision (C^2PS) *the algorithm that subdivides an* n_m*-valent polar configuration* $\mathbf{q}^m$ *to an* $2n_m$*-valent polar configuration* $\mathbf{q}^{m+1}$ *via (see Figure 4-1)*

$$\mathbf{q}_{00}^{m+1} := (1-\mathbb{a})\mathbf{q}_{00}^m + \frac{\mathbb{a}}{n_m}\sum_{\eta}^{\mathcal{G}_{\mathbf{q}^m}} \mathbf{q}_{1,[\eta]}^m = \frac{3}{4}\mathbf{q}_{00}^m + \frac{1}{4n_m}\sum_{\eta}^{\mathcal{G}_{\mathbf{q}^m}} \mathbf{q}_{1,[\eta]}^m \tag{4–1}$$

$$\begin{aligned}\mathbf{q}_{1,[\tau]}^{m+1} &:= (1-\hat{\mathbb{b}}_0)\mathbf{q}_{00}^m + \frac{1}{n_m}\sum_{\eta}^{\mathcal{G}_{\mathbf{q}^m}} \left(\hat{\mathbb{b}}_0 + c_{\eta-\tau} + \frac{1}{2}c_{2(\eta-\tau)}\right)\mathbf{q}_{1,[\eta]}^m \\ &= \frac{1}{2}\mathbf{q}_{00}^m + \frac{1}{n_m}\sum_{\eta}^{\mathcal{G}_{\mathbf{q}^m}} \left(\frac{1}{2} + c_{\eta-\tau} + \frac{1}{2}c_{2(\eta-\tau)}\right)\mathbf{q}_{1,[\eta]}^m \end{aligned} \tag{4–2}$$

$$\begin{aligned}\mathbf{q}_{2,[\tau]}^{m+1} &:= \mathbb{c}\tilde{\mathbf{q}}_{1,[\tau]}^m + (1-\mathbb{c})\tilde{\mathbf{q}}_{2,[\tau]}^m + \frac{2\hat{\mathbb{d}}_0}{n_m}\sum_{\eta}^{\mathcal{G}_{\mathbf{q}^m}} c_{\eta-\tau}\mathbf{q}_{1,[\eta]}^m \\ &= \frac{11}{12}\tilde{\mathbf{q}}_{1,[\tau]}^m + \frac{1}{12}\tilde{\mathbf{q}}_{2,[\tau]}^m - \frac{1}{6n_m}\sum_{\eta}^{\mathcal{G}_{\mathbf{q}^m}} c_{\eta-\tau}\mathbf{q}_{1,[\eta]}^m \end{aligned} \tag{4–3}$$

$$\begin{aligned}\mathbf{q}_{3,[\tau]}^{m+1} &:= \frac{1}{2}\tilde{\mathbf{q}}_{1,[\tau]}^m + \frac{1}{2}\tilde{\mathbf{q}}_{2,[\tau]}^m \qquad \mathbf{q}_{4,[\tau]}^{m+1} := \frac{1}{8}\tilde{\mathbf{q}}_{1,[\tau]}^m + \frac{6}{8}\tilde{\mathbf{q}}_{2,[\tau]}^m + \frac{1}{8}\tilde{\mathbf{q}}_{3,[\tau]}^m \\ \mathbf{q}_{5,[\tau]}^{m+1} &:= \frac{1}{2}\tilde{\mathbf{q}}_{2,[\tau]}^m + \frac{1}{2}\tilde{\mathbf{q}}_{3,[\tau]}^m \end{aligned} \tag{4–4}$$

where $\tilde{\mathbf{q}}^m$ *is obtained after subdividing* $\mathbf{q}^m$ *once in the circular direction. Observe that* $\mathbf{q}_{3,[\tau]}^{m+1}$, $\mathbf{q}_{4,[\tau]}^{m+1}$, $\mathbf{q}_{5,[\tau]}^{m+1}$ *are computed via uniform bi-3 spline subdivision. Let the operator* $\mathcal{T}$ *denote a single application of* C^2PS*. The limit surface* $\mathbf{x}^{\mathrm{C^2PS}}$ *is the union of spline rings* $G_m\mathbf{q}^m$ *where* $\mathbf{q}^m := \mathcal{T}^m(\mathbf{q}^0)$.

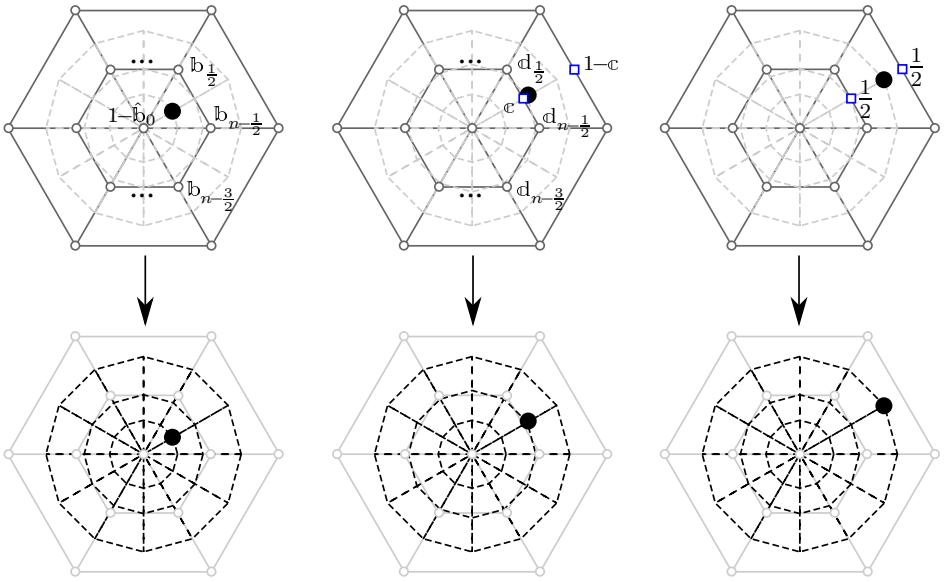


Figure 4-1. C^2 polar subdivision rules. The rules are the same as RTS (Figure 3-3), except that the outer i-links require intermediate uniform cubic circular subdivision (intermediate vertices indicated by □). The refined mesh (• and *dashed lines*) is computed as before from the old mesh (∘, *solid lines*) with $\mathbb{b}_j$ and $\mathbb{d}_j$ computed via (3–9) and (3–10) using half-integer indices. As in RTS, bi-3 rules are applied away from the polar vertex.

Since C^2PS subdivides in both the radial and circular directions simultaneously, it is directly compatible with Catmull-Clark, requiring no mesh separation for refinement: every quad on the coarse mesh yields four after subdivision, and each polar triangle splits into two polar triangles and two quads, as illustrated in Figure 4-1. Additionally, the limit surface is bi-3 and can be computed as a closed-form expression, which is difficult to do for RTS on an infinite-valent vertex. Each subsequent spline ring has twice as many polynomial patches and control points as its predecessor, and this exponentially-increasing order of approximation enables the spline ring sequence to converge to a second-order Taylor expansion at the pole.

4.2 Analysis

Since the connectivity is no longer stationary at the polar vertex, the traditional method of spectral analysis does not directly apply. However, if we rewrite (4–1)–(4–4) in eigenspace as we did RTS in Section 3.3.2, we can employ a similar analysis technique.

What helps is the intuition from RTS that the second-order expansion at the pole is determined by the eigensplines e_k multiplied by eigencoefficients $\mathbf{p}_k$ defined in (3–27).

The *eigencoefficient* $\mathbf{p}_k^m$ *of* C^2PS is also computed via (3–27) on $\mathbf{q}^m$, and we abbreviate $\mathbf{p}_k := \mathbf{p}_k^0$. The *eigensplines of* C^2PS are the limit surface $e_k^{\mathrm{C^2PS}} := \mathcal{L}^{\mathrm{C^2PS}}(\mathbf{v}_k)$ where $\mathbf{v}_k$ is an eigenvector of RTS with polar valence n_0. A superscript of RTS_{n_m} disambiguates the eigenspline $e_k^{\mathrm{RTS}_{n_m}}$ of RTS on a valence n_m polar configuration.

The following subsections will show the following.

- As in RTS, $\mathbf{p}_{k\in\mathbb{Z}_6}$ is preserved after every application of $\mathcal{T}$ – i.e. $\mathbf{p}_k^{m+1} = \ell_k \mathbf{p}_k^m = \ell_k^{m+1}\mathbf{p}_k$ (Lemma 5).
- $\mathcal{T}$ can be approximated in terms of $\mathbf{p}_{k\in\mathbb{Z}_6}$ plus a deviation of $\mathrm{O}\left(\frac{1}{8^m}\right)$ for polar valence n_m (Lemma 6).
- For $k \in \mathbb{Z}_{6n}$, $e_k^{\mathrm{C^2PS}}$ converges to $e_k^{\mathrm{RTS}_\infty}$ at the rate of $\mathrm{O}\left(\frac{1}{8^m}\right)$ at the pole (Lemma 7).
- The statements above yield a second-order Taylor expansion of $\mathbf{x}^{\mathrm{C^2PS}}$ at the pole proving that it is C^2 (Theorem 2).

4.2.1 Preservation of eigencoefficients

The following simplifications can be shown by using the addition rule for sine and cosine, and the orthogonality of the discrete Fourier basis.

$$\frac{1}{2n}\sum_g^{2n} c_{a_1(g-j):2n}\frac{1}{n}\sum_h^n c_{a_2(h-\frac{g}{2}):n}\mathbf{q}_{ih} = \begin{cases} \frac{1}{n}\sum_h^n \mathbf{q}_{ih} & \text{if } a_1 = a_2 = 0 \\ \frac{1}{2n}\sum_h^n c_{a_1(h-\frac{j}{2}):n}\mathbf{q}_{ih} & \text{if } a_1 = \pm a_2 \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\frac{1}{2n}\sum_g^{2n} s_{a_1(g-j):2n}\frac{1}{n}\sum_h^n c_{a_2(h-\frac{g}{2}):n}\mathbf{q}_{ih} = \begin{cases} \frac{\pm 1}{2n}\sum_h^n s_{a_1(h-\frac{j}{2}):n}\mathbf{q}_{ih} & \text{if } a_1 = \pm a_2 \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4–5}$$

Using these simplifications, we prove the following lemma for C^2PS.

Lemma 5. *For the subdivision algorithm* C^2PS, $\mathbf{p}_k^m = \ell_k^m \mathbf{p}_k$ *when* $k \in \mathbb{Z}_6$ *and* $m \geq 0$.

Proof. The base case $m = 0$ of the induction is trivially true. For the inductive step, we assume $\mathbf{p}_k^m = \ell_k^m \mathbf{p}_k$ and show that this property holds for $\mathbf{p}_k^{m+1}$ as well.

Case $k \in \{0, 3\}$:

$$\begin{aligned}
\mathbf{p}_3^{m+1} &\overset{(3\text{–}27)}{=} -\mathbf{q}_{00}^{m+1} + \frac{1}{n_{m+1}} \sum_{\gamma}^{\mathcal{G}_{\mathbf{q}^{m+1}}} \mathbf{q}_{1,[\gamma]}^{m+1} \\
&\overset{\substack{(4\text{–}1)\\(4\text{–}2)}}{=} -\left(\frac{3}{4}\mathbf{q}_{00}^{m} + \frac{1}{4n_m} \sum_{\eta}^{\mathcal{G}_{\mathbf{q}^{m}}} \mathbf{q}_{1,[\eta]}^{m}\right) \\
&\qquad + \frac{1}{2n_m} \sum_{\gamma}^{\mathcal{G}_{\mathbf{q}^{m+1}}} \left(\frac{1}{2}\mathbf{q}_{00}^{m} + \frac{1}{n_m} \sum_{\eta}^{\mathcal{G}_{\mathbf{q}^{m}}} \left(\frac{1}{2} + c_{\eta-\gamma} + \frac{1}{2}c_{2(\eta-\gamma)}\right) \mathbf{q}_{1,[\eta]}^{m}\right) \\
&\overset{(4\text{–}5)}{=} -\left(\frac{3}{4}\mathbf{q}_{00}^{m} + \frac{1}{4n_m} \sum_{\eta}^{\mathcal{G}_{\mathbf{q}^{m}}} \mathbf{q}_{1,[\eta]}^{m}\right) + \left(\frac{1}{2}\mathbf{q}_{00}^{m} + \frac{1}{n_m} \sum_{\eta}^{\mathcal{G}_{\mathbf{q}^{m}}} \frac{1}{2}\mathbf{q}_{1,[\eta]}^{m}\right) \\
&= \frac{1}{4}\left(-\mathbf{q}_{00}^{m} + \frac{1}{n_m} \sum_{\eta}^{\mathcal{G}_{\mathbf{q}^{m}}} \mathbf{q}_{1,[\eta]}^{m}\right) = \frac{1}{4}\mathbf{p}_3^m = \ell_3 \mathbf{p}_3^m = \ell_3^{m+1}\mathbf{p}_3
\end{aligned}$$

The sequence of steps for $k = 0$ is very similar to those of $k = 3$ above, and it similarly concludes that $\mathbf{p}_0^{m+1} = \ell_0^{m+1}\mathbf{p}_0$.

Case $k \in \{1, 2, 4, 5\}$:

$$\begin{aligned}
\mathbf{p}_1^{m+1} &\overset{(3\text{–}27)}{=} \frac{2}{n_{m+1}} \sum_{\gamma}^{\mathcal{G}_{\mathbf{q}^{m+1}}} c_\gamma \mathbf{q}_{1,[\gamma]}^{m+1} \\
&\overset{(4\text{–}2)}{=} \frac{2}{2n_m} \sum_{\gamma}^{\mathcal{G}_{\mathbf{q}^{m+1}}} c_\gamma \left(\frac{1}{2}\mathbf{q}_{00}^{m} + \frac{1}{n_m} \sum_{\eta}^{\mathcal{G}_{\mathbf{q}^{m}}} \left(\frac{1}{2} + c_{\eta-\gamma} + \frac{1}{2}c_{2(\eta-\gamma)} + \frac{1}{8}c_{3(\eta-\gamma)}\right) \mathbf{q}_{1,[\eta]}^{m}\right) \\
&\overset{\substack{(4\text{–}5)\ \&\\(c_\gamma = c_{0-\gamma})}}{=} \frac{2}{2n_m} \sum_{\eta}^{\mathcal{G}_{\mathbf{q}^{m}}} c_\eta \mathbf{q}_{1,[\eta]}^{m} = \frac{1}{2}\mathbf{p}_1^m = \ell_1 \mathbf{p}_1^m = \ell_1^{m+1}\mathbf{p}_1
\end{aligned}$$

The $k \in \{2, 4, 5\}$ cases are derived using a very similar sequence of steps, showing that for all six cases $\mathbf{p}_k^{m+1} = \ell_k^{m+1}\mathbf{p}_k$, completing the induction. □

4.2.2 Reformulation of C^2PS in terms of the eigencoefficients

In the same vein as (3–28)–(3–33), C^2PS can be reformulated to depend only on $\mathbf{p}_{k\in\mathbb{Z}_6}$ plus a deviation that diminishes quickly in the number of subdivisions.

Lemma 6. *The C^2PS refinement equations (4–1)–(4–4) are of the form*

$$\mathbf{q}_{00}^{m+1} = \mathbf{p}_0 - \frac{\mu^{m+1}}{3}\mathbf{p}_3 \tag{4–6}$$

$$\mathbf{q}_{1,[\tau]}^{m+1} = \mathbf{p}_0 + \lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{2\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) \tag{4–7}$$

$$\mathbf{q}_{2,[\tau]}^{m+1} = \mathbf{p}_0 + 2\lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{11\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right) \tag{4–8}$$

$$\mathbf{q}_{3,[\tau]}^{m+1} = \mathbf{p}_0 + 3\lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{26\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right) \tag{4–9}$$

$$\mathbf{q}_{4,[\tau]}^{m+1} = \mathbf{p}_0 + 4\lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{47\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right) \tag{4–10}$$

$$\mathbf{q}_{5,[\tau]}^{m+1} = \mathbf{p}_0 + 5\lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{74\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right) \tag{4–11}$$

Due to the interaction with circular subdivision, the derivation for the four outer links $\mathbf{q}_{2,[\tau]}^{m+1}$–$\mathbf{q}_{5,[\tau]}^{m+1}$ is involved and requires the introduction of new abstractions. The proof of (4–6)–(4–11) is hence deferred to the appendix to maintain the flow of this discussion. These equations can be reduced to

$$\begin{aligned}\mathbf{q}_{i,[\tau]}^{m+1} = {} & (\hat{\mathbf{v}}_0)_i \mathbf{p}_0 + (\hat{\mathbf{v}}_1)_i \lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) \\ & + \mu^{m+1}((\hat{\mathbf{v}}_3)_i \mathbf{p}_3 + (\hat{\mathbf{v}}_4)_i (\mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau})) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right),\end{aligned} \tag{4–12}$$

for $i \in \mathbb{Z}_6$, differing only by $\mathrm{O}\left(\frac{1}{8^{m+1}}\right)$ from (3–34) when the valences are equal.

4.2.3 Convergence of the eigensplines

Here we show that $e_k^{\mathrm{C^2PS}}(r,\tau)$ converges to $e_k^{\mathrm{RTS}_\infty}(r,\tau)$ as $r \to 0$.

Lemma 7. $\left|e_k^{\mathrm{C^2PS}}(r,\tau) - e_k^{\mathrm{RTS}_\infty}(r,\tau)\right|_{r\in[2\lambda^m, 4\lambda^m]} = \mathrm{O}\left(\frac{1}{8^m}\right)$

Proof. Since both RTS and C^2PS are affine-invariant, $e_0^{\mathrm{C^2PS}}(r,\tau) = e_0^{\mathrm{RTS}_\infty}(r,\tau) = 1$, and the lemma holds. Assume that $k > 0$, which also implies that $|\ell_k| \le \lambda$. Let $\mathbf{v}_k$ and $\tilde{\mathbf{v}}_k$ be the k^{th} eigenvectors of RTS of valence n_m and n_0, respectively.

$$E := \left|e_k^{\mathrm{C^2PS}}(r,\tau) - e_k^{\mathrm{RTS}_\infty}(r,\tau)\right|_{r\in[2\lambda^m, 4\lambda^m]} = \left|e_k^{\mathrm{C^2PS}}(\lambda^m r,\tau) - e_k^{\mathrm{RTS}_\infty}(\lambda^m r,\tau)\right|_{r\in[2,4]}$$

$$
\begin{aligned}
&\overset{\text{triangle inequality}}{\leq} \left| e_k^{\mathrm{C^2PS}}(\lambda^m r,\tau) - e_k^{\mathrm{RTS}_{n_m}}(\lambda^m r,\tau) \right|_{r\in[2,4]} + \left| e_k^{\mathrm{RTS}_{n_m}}(\lambda^m r,\tau) - e_k^{\mathrm{RTS}_\infty}(\lambda^m r,\tau) \right|_{r\in[2,4]} \\
&\overset{(3\text{–}3)}{=} \underbrace{\left| G_m \mathcal{T}^m(\tilde{\mathbf{v}}_k)(r,\tau) - G_m A^m(\mathbf{v}_k)(r,\tau) \right|}_{\mathcal{T}^m(\tilde{\mathbf{v}}_k) \text{ and } A^m(\mathbf{v}_k) \text{ have valence } n_m;\ G_m \text{ is linear}} + \left| \ell_k^m e_k^{\mathrm{RTS}_{n_m}}(r,\tau) - \ell_k^m e_k^{\mathrm{RTS}_\infty}(r,\tau) \right|_{r\in[2,4]} \\
&= \underbrace{\left| G_m(\mathcal{T}^m(\tilde{\mathbf{v}}_k) - A^m(\mathbf{v}_k))(r,\tau) \right|}_{E_1 :=} + \underbrace{\ell_k^m}_{\mathrm{O}(\lambda^m)} \underbrace{\left| e_k^{\mathrm{RTS}_{n_m}}(r,\tau) - e_k^{\mathrm{RTS}_\infty}(r,\tau) \right|_{r\in[2,4]}}_{\text{Lemma 4} \Rightarrow\ =\mathrm{O}\left(\frac{1}{n_m^2}\right)=\mathrm{O}\left(\frac{1}{4^m}\right)} \\
&= E_1 + \mathrm{O}\left(\frac{1}{8^m}\right)
\end{aligned}
$$

By definition, $\mathbf{p}_h = \delta_{hk}$ for $h \in \mathbb{Z}_6$ when computed on either eigenvector $\tilde{\mathbf{v}}_k$ or $\mathbf{v}_k$; in other words, the first six eigencoefficients of these two eigenvectors match. Since (4–12) and (3–34) define $\mathcal{T}^m(\tilde{\mathbf{v}}_k)$ and $A^m(\mathbf{v}_k)$, respectively, in terms of $\mathbf{p}_{h\in\mathbb{Z}_6}$, and these two formulae differ by $\mathrm{O}\left(\frac{1}{8^m}\right)$, it follows that $\mathcal{T}^m(\tilde{\mathbf{v}}_k) - A^m(\mathbf{v}_k) = \mathrm{O}\left(\frac{1}{8^m}\right)$. Therefore, $E_1 = \mathrm{O}\left(\frac{1}{8^m}\right)$, and $E = \mathrm{O}\left(\frac{1}{8^m}\right)$, proving the lemma. □

4.2.4 Proof of curvature continuity

We can now establish a second-order Taylor expansion at the pole, proving curvature continuity.

Theorem 2. *C^2PS is C^2 at the pole.*

Proof. Recall from (3–24) that a polar configuration $\mathbf{q}^0$ of valence n_0 can be written as the following linear combination of the eigenvectors, $\mathbf{v}_k^0$, $k \in \mathbb{Z}_{6n_0}$.

$$
\begin{aligned}
\mathbf{q}_{ij}^0 = \sum_k^{6n_0} \mathbf{p}_k \, (\mathbf{v}_k^0)_{ij} \quad &\Rightarrow \quad \mathcal{L}^{\mathrm{C^2PS}}(\mathbf{q}^0) = \sum_k^{6n_0} \mathbf{p}_k \, \mathcal{L}^{\mathrm{C^2PS}}(\mathbf{v}_k^0) \\
\Rightarrow \mathbf{x}^{\mathrm{C^2PS}}(r,\tau) = \sum_k^{6n_0} \mathbf{p}_k \, e_k^{\mathrm{C^2PS}}(r,\tau)
\end{aligned}
$$

We examine the sequence $\mathbf{x}^{\mathrm{C^2PS}}(r,\tau)\big|_{r\in[2\lambda^m,4\lambda^m]} \overset{(3\text{–}3)}{=} \left(G_m \mathcal{T}^m \mathbf{q}^0\right)(r,\tau)$ of spline rings that approach the pole. Since $r \in [2\lambda^m, 4\lambda^m] \Rightarrow \frac{r}{\lambda^m} \in [2,4]$, $\frac{r}{\lambda^m}$ is bounded away from 0 and ∞ and has no impact on asymptotic behavior when multiplied. This simplifies

$\mathrm{O}\left(\frac{1}{8^m}\right) = \mathrm{O}\left(\lambda^{3m}\right) = \mathrm{O}\left(\left(\frac{r}{\lambda^m}\right)^3 \lambda^{3m}\right) = \mathrm{O}\left(r^3\right)$. Thus,

$$
\begin{aligned}
\mathbf{x}^{\mathrm{C^2PS}}(r,\tau)\Big|_{r\in[2\lambda^m,4\lambda^m]} &= \sum_k^{6n_0} \mathbf{p}_k\, e_k^{\mathrm{C^2PS}}(r,\tau)\Big|_{r\in[2\lambda^m,4\lambda^m]} \\
\text{Lemma 7} &= \sum_k^{6n_0} \left(\mathbf{p}_k\, e_k^{\mathrm{RTS}\infty}(r,\tau) + \underbrace{\mathrm{O}\left(\frac{1}{8^m}\right)}_{\mathrm{O}(r^3)} \right)\Bigg|_{r\in[2\lambda^m,4\lambda^m]} \\
&= \Bigg(\mathbf{p}_0\, e_0^{\mathrm{RTS}\infty}(r,\tau) + \left(\mathbf{p}_1\, e_1^{\mathrm{RTS}\infty}(r,\tau) + \mathbf{p}_2\, e_2^{\mathrm{RTS}\infty}(r,\tau)\right) \\
&\quad + \left(\mathbf{p}_3\, e_3^{\mathrm{RTS}\infty}(r,\tau) + \mathbf{p}_4\, e_4^{\mathrm{RTS}\infty}(r,\tau) + \mathbf{p}_5\, e_5^{\mathrm{RTS}\infty}(r,\tau)\right) \\
&\quad + \sum_{k=6}^{6n_0} \mathbf{p}_k\, e_k^{\mathrm{RTS}\infty}(r,\tau) + \mathrm{O}\left(r^3\right) \Bigg)\Bigg|_{r\in[2\lambda^m,4\lambda^m]} \\
\text{Lemma 2 \& Lemma 4} &= \mathbf{p}_0 + r\left(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau\right) + r^2\left(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}\right) + \mathrm{o}\left(r^2\right)\Big|_{r\in[2\lambda^m,4\lambda^m]}
\end{aligned}
$$

Changing to Cartesian coordinates $(x,y) := (rc_\tau, rs_\tau)$, $\overline{\mathbf{x}}^{\mathrm{C^2PS}}(x,y) := \mathbf{x}^{\mathrm{C^2PS}}(r,\tau)$,

$$
\begin{aligned}
&\mathbf{x}^{\mathrm{C^2PS}}(r,\tau)\Big|_{r\in[2\lambda^m,4\lambda^m]} = \overline{\mathbf{x}}^{\mathrm{C^2PS}}(x,y)\Big|_{\sqrt{x^2+y^2}\in[2\lambda^m,4\lambda^m]} \\
&= \left(\mathbf{p}_0 + (\mathbf{p}_1 x + \mathbf{p}_2 y) + \left(\mathbf{p}_3(x^2+y^2) + \mathbf{p}_4(x^2-y^2) + \mathbf{p}_5(2xy)\right) + \mathrm{o}\left(x^2+y^2\right)\right)\Big|_{\sqrt{x^2+y^2}\in[2\lambda^m,4\lambda^m]},
\end{aligned}
$$

giving an explicit second-order expansion at the pole when $m \to \infty$. Hence the construction is C^2. □

The explicit Taylor expansion at the pole allows one to compute principal curvatures and directions. In some constructions, curvature continuity comes at the cost of macroscopic shape deterioration, even though the microscopic shape is improved. Chapter 5 shows empirically that our construction does not suffer from this defect; it generates surfaces of high visual quality.

CHAPTER 5
RESULTS AND DISCUSSION

Figure 5-1 shows a side-by-side comparison of RTS, RTS_∞, and C^2PS. To avoid curvature fluctuations in the first and second spline rings (Figure 5-1(A-B)), uniform (bi)cubic subdivision is applied to compute the refined 2-link on the first subdivision step (Figure 5-1(C-E)). For RTS, this is equivalent to applying bicubic polar subdivision on the first radial subdivision step, while using RTS on all the subsequent ones. The n-sidedness of the RTS curvature distribution is obvious, while RTS_∞ and C^2PS yield smoother curvature transitions in the circular direction. C^2PS distributes curvature more evenly, resulting in a lower maximal Gaussian curvature than RTS or RTS_∞ near the pole. As expected, for higher valences, the limit surfaces of these three algorithms are similar (Figure 5-2). Figure 5-3 tests C^2PS against various challenging configurations. The smooth highlight lines attest to the surface quality in the vicinity of the pole, even on higher-order saddles.

It may be possible to devise a bi-degree-4 C^3 polar scheme using a similar technique. The key ingredient is that the spline rings constituting the limit surface would need to shrink more rapidly to the pole. To see this, observe that the reformulation of C^2PS in terms of eigencoefficients (Lemma 6) is proved by simplifying the treatment of arbitrary number of circular subdivisions using a parameterized equivalence class $\text{aff}^4_{[\gamma]}(\mathbf{q}_i)$. The use of this class contributes a deviation of $\text{O}\left(\frac{1}{8^m}\right)$, which is the product of λ and the convergence rate $\text{O}\left(\frac{1}{n_m^2}\right) = \text{O}\left(\frac{1}{4^m}\right)$ of piecewise linear approximations to cosines and sines. Section 4.2.4 showed that for $\ell_1 = \ell_2 = \lambda = \frac{1}{2}$, $\text{O}\left(\frac{1}{8^m}\right)$ simplifies to $\text{O}\left(\lambda^{3m}\right) = \text{O}(r^3)$, contributing to the third-order term of the Taylor expansion at the pole. While our C^2 algorithm is unaffected by this, designing a C^3 algorithm requires understanding the third-order term precisely. One way to employ the simplicity of our proofs is to enforce eigenvalues $\ell_1 = \ell_2 = \lambda < \frac{1}{2}$, resulting in a deviation of $\text{O}\left(\frac{\lambda^m}{4^m}\right) = \text{o}\left(\lambda^{3m}\right)$ that converges to 0 more quickly than r^3, avoiding interference with the third-order expansion.

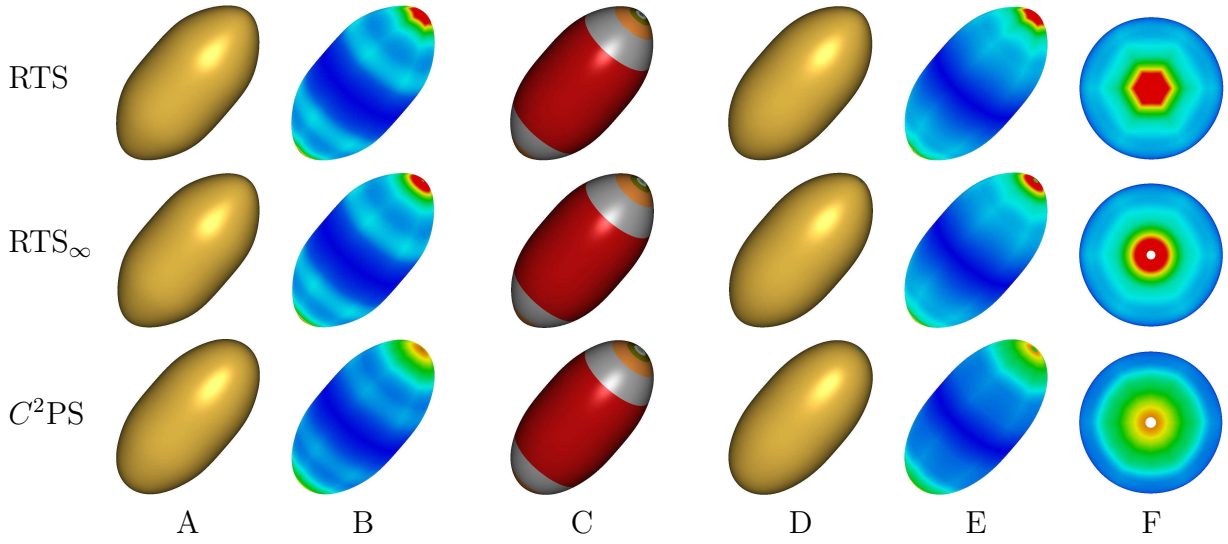


Figure 5-1. Comparison of RTS, RTS$_\infty$, and C^2PS. (C) spline rings defining limit surfaces (A and D) and Gaussian curvature (B and E) of two different initialization strategies of each scheme on Figure 2-5B input. Direct application produces (A) a sharper bend in the silhouette and (B) an abrupt curvature transition (*dark blue* means zero Gauss curvature), whereas using bicubic subdivision to compute the 2-link for the first subdivision step improves the curvature distribution (C, D, E). (F) RTS reveals an n-sidedness in its curvature distribution, while the curvature of RTS$_\infty$ and C^2PS is much more symmetric.

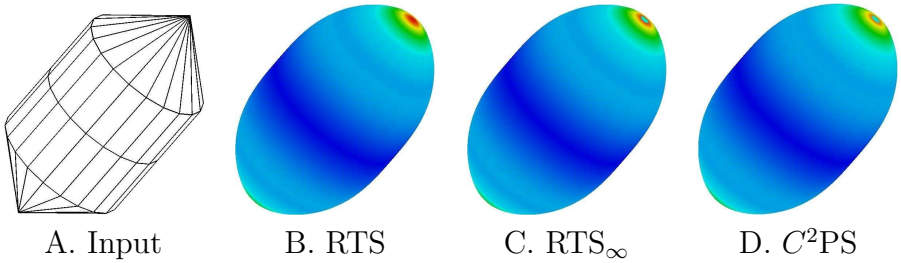


Figure 5-2. RTS, RTS$_\infty$, and C^2PS on a polar configuration of valence 20 show that their limit surfaces and Gaussian curvature distributions (*dark blue* is zero curvature) are similar for large polar valences.

Catmull-Clark extraordinary vertices of arbitrary valence can be converted to polar configurations, as demonstrated in Figure 5-4, additionally creating 5-valent extraordinary vertices (Figure 5-4A) or pentagons (Figure 5-4C). Thus, it may be possible to devise C^2 algorithms for either 5-valent extraordinary vertices or pentagons to construct surfaces that are globally C^2. This is left for future work.

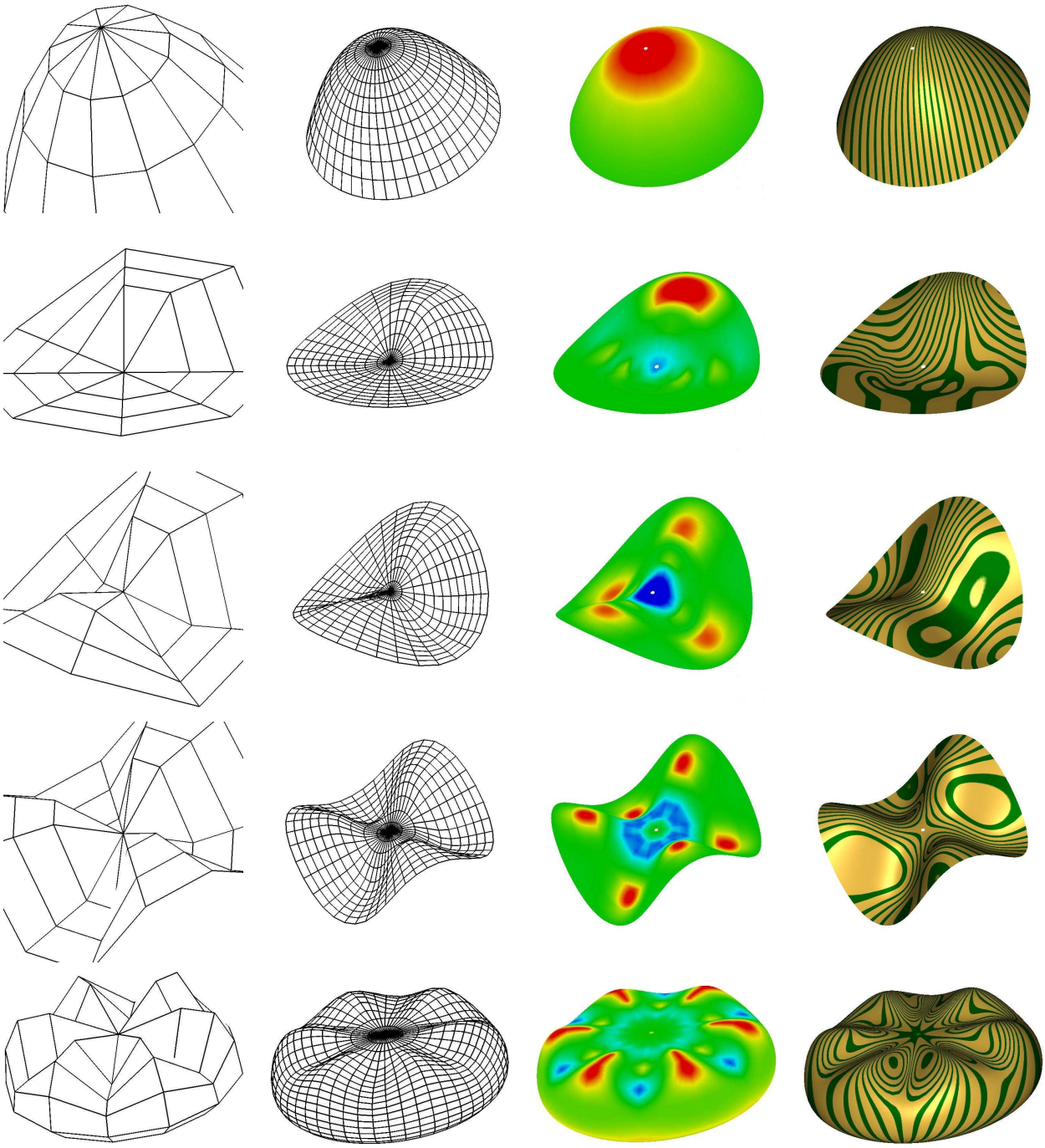


Figure 5-3. Shape gallery demonstrating that C^2PS performs with good shape. A) Input, B) twice subdivided mesh, C) Gaussian curvature of limit surface, and D) highlight lines. Zero curvature is *green*, while negative curvature is *blue* and positive is *red*.

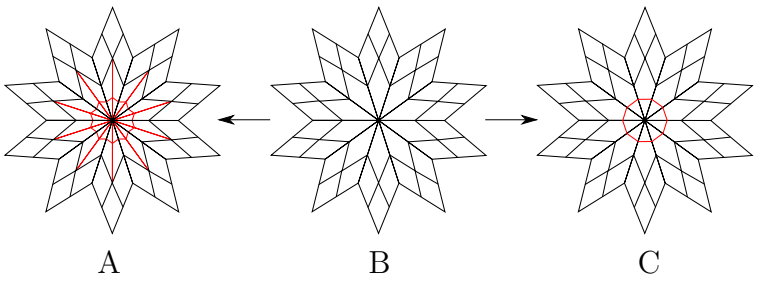


Figure 5-4. B) An n-valent Catmull-Clark extraordinary vertex can be converted to either A) a $2n$-valent polar configuration and n 5-valent extraordinary vertices, or C) a n-valent polar configuration and n pentagons.

CHAPTER 6
CONCLUSION

For quad meshes, we have introduced the polar configuration, which appears naturally at the ends of elongated objects, like the nose of a plane or the tips of fingers, where control lines along the same tensor direction meet to form a singularity. We have presented three polar subdivision algorithms compatible with Catmull-Clark [Catmull and Clark, 1978] subdivision: RTS, RTS$_\infty$, and C^2PS. While RTS surfaces are only C^1 with bounded curvature at its pole, RTS$_\infty$ and C^2PS have been shown to be fully C^2. And while the second-order continuity of RTS$_\infty$ is easier to prove, this algorithm transitions from a polynomial spline boundary to a non-polynomial surface that is, in general, not easy to compute exactly. Moreover, as a mesh refinement algorithm, RTS$_\infty$ is more complex to implement, requiring a logical separation of the polar configuration from the rest of the input mesh before subdivision is applied to it. In contrast, the entirely spline-based C^2PS is simpler both as a mesh refinement algorithm, and for explicitly evaluating the limit surface. However, since C^2PS results in *non-stationary* connectivity, standard subdivision theory fails to apply, and the proof of curvature-continuity at the pole is more complex. Nevertheless, we have shown, in this study that the algorithm is C^2 and given evidence that it tends to give good shape.

Subdivision algorithms are an accepted standard in animation and are sometimes used for conceptual design in CAD. These algorithms have been been avoided for high-quality surfaces in CAD partially due to shape problems near extraordinary vertices. We have gone one step closer to show that a subdivision algorithm may not be complex and still have good shape if non-stationary connectivity can be exploited to increase the order of approximation in the vicinity of the pole. We offer an additional incentive to use our method because theory developed in [Reif, 1998] and [Myles et al., 2008] suggests that curvature continuity may require a degree 6 NURBS surface when more than 4 NURBS meet at a point. On the other hand, we have shown that degree 3 is sufficient for a simple

subdivision algorithm exploiting non-stationary connectivity. We hope techniques such as ours help make subdivision surfaces more useful in mainstream CAD.

APPENDIX: C^2PS IN TERMS OF THE EIGENCOEFFICIENTS

Here, we derive in detail the reformulation (4–6)–(4–11) of C^2PS. A checkmark (✔) indicates that one of these equations has been proved. While $\mathbf{q}_{00}^{m+1}$ and $\mathbf{q}_{1j}^{m+1}$ are readily expressed in terms of the eigencoefficients,

$$\begin{aligned}
\mathbf{q}_{00}^{m+1} &= \frac{3}{4}\mathbf{q}_{00}^{m} + \frac{1}{4n_m}\sum_{\eta}^{\mathcal{G}\mathbf{q}^m}\mathbf{q}_{1,[\eta]}^{m} \quad = \quad \mathbf{p}_0^m - \frac{1}{12}\mathbf{p}_3^m \\
\boxed{\checkmark} &= \mathbf{p}_0 - \frac{\mu^m}{12}\mathbf{p}_3 \quad = \quad \mathbf{p}_0 - \frac{\mu^{m+1}}{3}\mathbf{p}_3 \qquad \text{(A–1)} \\
\mathbf{q}_{1,[\tau]}^{m+1} &= \frac{1}{2}\mathbf{q}_{00}^{m} + \frac{1}{n_m}\sum_{\eta}^{\mathcal{G}\mathbf{q}^m}\left(\frac{1}{2} + c_{\eta-\tau} + \frac{1}{2}c_{2(\eta-\tau)} + \frac{1}{8}c_{3(\eta-\tau)}\right)\mathbf{q}_{1,[\eta]}^{m} \\
&\overset{\text{addition rule for cosine \& (3–27)}}{=} \mathbf{p}_0^m + \frac{1}{2}(\mathbf{p}_1^m c_\tau + \mathbf{p}_2^m s_\tau) + \frac{1}{6}(\mathbf{p}_3^m + \mathbf{p}_4^m c_{2\tau} + \mathbf{p}_5^m s_{2\tau}) \\
\boxed{\checkmark} &\overset{\text{Lemma 5}}{=} \mathbf{p}_0 + \lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{2\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) \qquad \text{(A–2)} \\
\mathbf{q}_{2,[\tau]}^{m+1} &\overset{\text{(4–3) \& addition for cosine \& (3–27)}}{=} \frac{11}{12}\tilde{\mathbf{q}}_{\tau,[1]}^{m} + \frac{1}{12}\tilde{\mathbf{q}}_{\tau,[2]}^{m} - \frac{1}{6}(\mathbf{p}_1^m c_\tau + \mathbf{p}_2^m s_\tau) \\
&\overset{\text{Lemma 5}}{=} \frac{11}{12}\tilde{\mathbf{q}}_{\tau,[1]}^{m} + \frac{1}{12}\tilde{\mathbf{q}}_{\tau,[2]}^{m} - \frac{\lambda^{m+1}}{3}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau), \qquad \text{(A–3)}
\end{aligned}$$

the expressions for the four outer links $\mathbf{q}_2$ (A–3), $\mathbf{q}_3$, $\mathbf{q}_4$, and $\mathbf{q}_5$ are involved due to circular subdivision, and only the dominant terms will be shown and needed. With the intuition that every point on a spline is an affine (in fact, convex) combination of the four B-spline control points that are parametrically closest to it, we define the following equivalence class of affine combinations.

Definition 3 ($\mathrm{aff}^4_{[\gamma]}$)**.** *Let* $\mathbf{u}$ *be a vector of* n *B-spline control points of a periodic uniform cubic spline with knot sequence* $\frac{1}{n}\mathbb{Z}_n$*. The control points* $\mathbf{u} = \left[\mathbf{u}_{\left[\frac{0}{n}\right]}, \ldots, \mathbf{u}_{\left[\frac{n-1}{n}\right]}\right]$ *are indexed by their Greville abscissae so that adjacent pairs of Greville abscissae are* $\frac{1}{n}$ *apart. The equivalence class* $\mathrm{aff}^4_{[\gamma]}(\mathbf{u})$ *of all local affine combinations centered at* γ *is defined as*

$$\mathrm{aff}^4_{[\gamma]}(\mathbf{u}) := \left\{ \sum_{g}^{4} u_g \mathbf{u}_{[\gamma_g]} \;\middle|\; \begin{array}{l} \sum_g^4 u_g = 1, \;\; \sum_g^4 u_g\gamma_g = \gamma, \;\; \textit{and } \mathbf{u}_{[\gamma_0]}, \ldots, \mathbf{u}_{[\gamma_3]} \textit{ are the four} \\ \textit{control points whose Greville abscissae } \gamma_g \textit{ are closest to} \\ \gamma, \textit{ or have weight } u_g = 0 \textit{ if they tie for fourth place.} \end{array} \right\}$$

Since adjacent Greville abscissae differ by $\frac{1}{n}$, the affine combinations in $\mathrm{aff}^4_{[\gamma]}(\mathbf{u})$ are such that $|\gamma_g - \gamma| < \frac{2}{n} = \mathrm{O}(\frac{1}{n})$ if $u_g \neq 0$. Cubic B-spline refinement rules ensure that all affine combinations resulting from arbitrarily-many cubic B-spline refinements on $\mathbf{u}$ belong in $\mathrm{aff}^4_{[\gamma]}(\mathbf{u})$. The control points in each circularly-subdivided i-link $\tilde{\mathbf{q}}_i^m$ are affine combinations of control points in the i-link $\mathbf{q}_i^m$. Since these affine combinations result in an arbitrary number of cases in our reformulation of C^2PS, we focus on simplifying affine combinations of trigonometric functions.

Lemma 8. *If* $\mathbf{u} = [\mathrm{op}_k(\frac{g}{n})]_{g\in\mathbb{Z}_n}$, *then for all* $\tilde{\mathbf{u}}_{[\gamma]} := \sum_g^4 u_g \mathbf{u}_{[\gamma_g]} \in \mathrm{aff}^4_{[\gamma]}(\mathbf{u})$, $\tilde{\mathbf{u}}_{[\gamma]} = \mathrm{op}_k(\gamma) + \mathrm{O}\left(\frac{\alpha_k^2}{n^2}\right)$.

Proof.

Case 1: $\alpha_k = 0$ (i.e. $\mathrm{op}_k(\gamma) = c_{0\gamma} = 1$)

For $\mathbf{u} = [1]_{g\in\mathbb{Z}_n}$, $\tilde{\mathbf{u}}_{[\gamma]} = \sum_g^4 u_g \underbrace{\mathbf{u}_{[\gamma_g]}}_{1} = \sum_g^4 u_g = 1 = \mathrm{op}_k(\gamma)$.

Case 2: $\mathbf{u}_\gamma = c_{\alpha_k\gamma}$, $\alpha_k \neq 0$

$$\begin{aligned}
\tilde{\mathbf{u}}_{[\gamma]} &= \sum_g^4 u_g c_{\alpha_k\gamma_g} \quad = \quad \sum_g^4 u_g c_{\alpha_k(\gamma_g-\gamma)+\alpha_k\gamma} \quad = \quad \sum_g^4 u_g \left(c_{\alpha_k(\gamma_g-\gamma)}c_{\alpha_k\gamma} - s_{\alpha_k(\gamma_g-\gamma)}s_{\alpha_k\gamma}\right) \\
&= \sum_g^4 u_g \underbrace{\left(1 + \mathrm{O}\left(\frac{\alpha_k^2}{n^2}\right)\right)}_{\text{from } c_{\alpha_k(\gamma_g-\gamma)}} c_{\alpha_k\gamma} - u_g \underbrace{\left(\alpha_k(\gamma_g-\gamma) + \mathrm{O}\left(\frac{\alpha_k^3}{n^3}\right)\right)}_{\text{from } s_{\alpha_k(\gamma_g-\gamma)}} s_{\alpha_k\gamma} \qquad \begin{array}{l}\text{Taylor expan. \&} \\ |\gamma_g-\gamma| = \mathrm{O}\left(\frac{1}{n}\right)\end{array} \\
&= c_{\alpha_k\gamma} \cancelto{1}{\sum_g^4 u_g} + \alpha_k s_{\alpha_k\gamma} \cancelto{0}{\sum_g^4 u_g(\gamma_g-\gamma)} + \mathrm{O}\left(\frac{\alpha_k^2}{n^2}\right) \quad = \quad c_{\alpha_k\gamma} + \mathrm{O}\left(\frac{\alpha_k^2}{n^2}\right),
\end{aligned}$$

satisfying the theorem.

Case 3: $\mathbf{u}_\gamma = -s_{\alpha_k\gamma}$, $\alpha_k \neq 0$

The proof is almost identical to Case 2 and shows that $\tilde{\mathbf{u}}_{[\gamma]} = s_{\alpha_k\gamma} + \mathrm{O}\left(\frac{\alpha_k^2}{n^2}\right)$, satisfying the theorem. □

Equipped with Lemma 8, we can now estimate $\mathbf{q}^{m+1}_{2,[\tau]}$ by describing $\mathrm{aff}^4_{[\tau]}(\mathbf{q}_2^{m+1}) \ni \mathbf{q}^{m+1}_{2,[\tau]}$ in terms of $\mathbf{p}_{k\in\mathbb{Z}_6}$. The bound $\mathrm{O}\left(\frac{\alpha_k^2}{n_m^2}\right)$ on the terms not explicitly written in terms of $\mathbf{p}_k$ simplifies to $\mathrm{O}\left(\frac{1}{4^m}\right)$ since $\alpha_k \in \{0,1,2\}$ in the relevant cases and $n_m = n_0 2^m$. For each

$\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_1^{m+1}} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_1^{m+1})$,

$$\begin{aligned}
\text{(A–2) \& Lemma 8}\quad \tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_1^{m+1}} &= \mathbf{p}_0 + \lambda^{m+1}\left(\mathbf{p}_1\left(c_\tau + \mathrm{O}\left(\frac{1}{4^m}\right)\right) + \mathbf{p}_2\left(s_\tau + \mathrm{O}\left(\frac{1}{4^m}\right)\right)\right) \\
&\quad + \frac{2\mu^{m+1}}{3}\left(\mathbf{p}_3 + \mathbf{p}_4\left(c_{2\tau} + \mathrm{O}\left(\frac{1}{4^m}\right)\right) + \mathbf{p}_5\left(s_{2\tau} + \mathrm{O}\left(\frac{1}{4^m}\right)\right)\right) \\
&= \mathbf{p}_0 + \lambda^{m+1}\left(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau\right) + \frac{2\mu^{m+1}}{3}\left(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}\right) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right) \qquad \text{(A–4)}
\end{aligned}$$

For each $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^{m+1}} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_2^{m+1})$, there exist $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_1^m} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_1^m)$ and $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^m} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_2^m)$ so that

$$\begin{aligned}
\text{(A–3) \& Lemma 8}\quad \tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^{m+1}} &= \frac{11}{12}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_1^m} + \frac{1}{12}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^m} - \frac{\lambda^{m+1}}{3}\left(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau + \mathrm{O}\left(\frac{1}{4^m}\right)\right) \\
\text{(A–4) \& Lemma 8}\quad &= \frac{11}{12}\left(\mathbf{p}_0 + \lambda^m\left(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau\right) + \frac{2\mu^m}{3}\left(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}\right) + \mathrm{O}\left(\frac{1}{8^m}\right)\right) \\
&\quad + \frac{1}{12}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^m} - \frac{\lambda^{m+1}}{3}\left(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau\right) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right) \\
&= \frac{11}{12}\mathbf{p}_0 + \frac{5}{3}\lambda^{m+1}\left(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau\right) + \frac{22}{9}\mu^{m+1}\left(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}\right) \\
&\quad + \mathrm{O}\left(\frac{1}{8^{m+1}}\right) + \frac{1}{12}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^m} \qquad \text{(A–5)}
\end{aligned}$$

(A–5) describes the set $\mathrm{aff}_{[\tau]}^4(\mathbf{q}_2^{m+1})$ recursively with respect to m. Expanding out the recursion shows that for each $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^{m+1}} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_2^{m+1})$, there exists $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^0} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_2^0)$ so that

$$\begin{aligned}
\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^{m+1}} &= \frac{11}{12}\left(\sum_h^{m+1}\frac{1}{12^h}\right)\mathbf{p}_0 + \frac{5}{3}\left(\sum_h^{m+1}\frac{\lambda^{m+1-h}}{12^h}\right)\left(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau\right) \\
&\quad + \frac{22}{9}\left(\sum_h^{m+1}\frac{\mu^{m+1-h}}{12^h}\right)\left(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}\right) + \mathrm{O}\left(\sum_h^{m+1}\frac{1}{8^{m+1-h}12^h}\right) + \frac{1}{12^{m+1}}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^0} \\
\text{geom. series}\quad &= \left(1 - \frac{1}{12^{m+1}}\right)\mathbf{p}_0 + 2\lambda^{m+1}\left(1 - \frac{1}{6^{m+1}}\right)\left(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau\right) \\
&\quad + \mu^{m+1}\frac{11}{3}\left(1 - \frac{1}{3^{m+1}}\right)\left(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}\right) \\
&\quad + \mathrm{O}\left(\frac{3}{8^{m+1}}\left(1 - \left(\frac{2}{3}\right)^{m+1}\right)\right) + \frac{1}{12^{m+1}}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^0} \\
\boxed{\checkmark} &= \mathbf{p}_0 + 2\lambda^{m+1}\left(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau\right) + \frac{11\mu^{m+1}}{3}\left(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}\right) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right) \qquad \text{(A–6)}
\end{aligned}$$

Since $\mathbf{q}_{2,[\tau]}^{m+1} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_2^{m+1})$, it too is described by (A–6), proving 4–8. We similarly derive formulas for $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_3^{m+1}} \ni \mathrm{aff}_{[\tau]}^4(\mathbf{q}_3^{m+1})$, $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_4^{m+1}} \ni \mathrm{aff}_{[\tau]}^4(\mathbf{q}_4^{m+1})$, and $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_5^{m+1}} \ni \mathrm{aff}_{[\tau]}^4(\mathbf{q}_5^{m+1})$, automatically yielding formulas for $\mathbf{q}_{3,[\tau]}^{m+1}$, $\mathbf{q}_{4,[\tau]}^{m+1}$, and $\mathbf{q}_{5,[\tau]}^{m+1}$. For each $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_3^{m+1}} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_3^{m+1})$, there exist $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_1^m} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_1^m)$ and $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^m} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_2^m)$ so that

$$\begin{aligned}
\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_3^{m+1}} &= \frac{1}{2}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_1^{m+1}} + \frac{1}{2}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^{m+1}} \\
\overset{\text{(A–2)}}{\underset{\text{(A–6)}}{=}} & \frac{1}{2}\left(\mathbf{p}_0 + \lambda^m(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{2\mu^m}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^m}\right)\right) \\
& + \frac{1}{2}\left(\mathbf{p}_0 + 2\lambda^m(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{11\mu^m}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^m}\right)\right) \\
\boxed{\checkmark} &= \mathbf{p}_0 + 3\lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{26\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right)
\end{aligned} \tag{A–7}$$

For each $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_4^{m+1}} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_4^{m+1})$, there exist $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_1^m} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_1^m)$, $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^m} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_2^m)$, and $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_3^m} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_3^m)$ so that

$$\begin{aligned}
\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_4^{m+1}} &= \frac{1}{8}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_1^{m+1}} + \frac{6}{8}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^{m+1}} + \frac{1}{8}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_3^{m+1}} \\
\overset{\text{(A–2), (A–6)}}{\underset{\text{\& (A–7)}}{=}} & \frac{1}{8}\left(\mathbf{p}_0 + \lambda^m(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{2\mu^m}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau})\right) \\
& + \frac{6}{8}\left(\mathbf{p}_0 + 2\lambda^m(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{11\mu^m}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^m}\right)\right) \\
& + \frac{1}{8}\left(\mathbf{p}_0 + 3\lambda^m(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{26\mu^m}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^m}\right)\right) \\
\boxed{\checkmark} &= \mathbf{p}_0 + 4\lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{47\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right)
\end{aligned} \tag{A–8}$$

For each $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_5^{m+1}} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_5^{m+1})$, there exist $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^m} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_2^m)$ and $\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_3^m} \in \mathrm{aff}_{[\tau]}^4(\mathbf{q}_3^m)$ so that

$$\begin{aligned}
\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_5^{m+1}} &= \frac{1}{2}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_2^{m+1}} + \frac{1}{2}\tilde{\mathbf{u}}_{[\tau]}^{\mathbf{q}_3^{m+1}} \\
\overset{\text{(A–6)}}{\underset{\text{(A–7)}}{=}} & \frac{1}{2}\left(\mathbf{p}_0 + 2\lambda^m(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{11\mu^m}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^m}\right)\right) \\
& + \frac{1}{2}\left(\mathbf{p}_0 + 3\lambda^m(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{26\mu^m}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^m}\right)\right) \\
\boxed{\checkmark} &= \mathbf{p}_0 + 5\lambda^{m+1}(\mathbf{p}_1 c_\tau + \mathbf{p}_2 s_\tau) + \frac{74\mu^{m+1}}{3}(\mathbf{p}_3 + \mathbf{p}_4 c_{2\tau} + \mathbf{p}_5 s_{2\tau}) + \mathrm{O}\left(\frac{1}{8^{m+1}}\right)
\end{aligned} \tag{A–9}$$

REFERENCES

Augsdörfer, U. H., Dodgson, N. A., and Sabin, M. A. 2006. Tuning subdivision by minimising Gaussian curvature variation near extraordinary vertices. *Computer Graphics Forum (Proceedings of Eurographics) 25,* 3, 263–272.

Boier-Martin, I. and Zorin, D. 2004. Differentiable parameterization of Catmull-Clark subdivision surfaces. In *SGP '04: Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*. ACM Press, New York, NY, USA, 155–164.

Catmull, E. and Clark, J. 1978. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer Aided Design 10*, 350–355.

Doo, D. and Sabin, M. 1978. Behaviour of recursive division surfaces near extraordinary points. *Computer Aided Design 10*, 356–360.

Dyn, N., Levin, D., and Gregory, J. A. 1990. A butterfly subdivision scheme for surface interpolation with tension control. *ACM Transactions on Graphics 9,* 2, 160–169.

Ginkel, I. and Umlauf, G. 2006. Loop subdivision with curvature control. In *SGP '06: Proceedings of the Fourth Eurographics Symposium on Geometry Processing*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 163–171.

Karčiauskas, K., Myles, A., and Peters, J. 2006. A C^2 polar jet subdivision. In *SGP '06: Proceedings of the Fourth Eurographics Symposium on Geometry Processing*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 173–180.

Karčiauskas, K. and Peters, J. 2007. Surfaces with polar structure. *Computing 79,* 2, 309–315.

Karčiauskas, K., Peters, J., and Reif, U. 2004. Shape characterization of subdivision surfaces – case studies. *Computer Aided Geometric Design 21,* 6 (July), 601–614.

Karčiauskas, K. and Peters, J. 2007a. Bicubic polar subdivision. *ACM Transactions on Graphics 26,* 4, 14.

Karčiauskas, K. and Peters, J. 2007b. Concentric tessellation maps and curvature continuous guided surfaces. *Computer Aided Geometric Design 24,* 2, 99–111.

Karčiauskas, K. and Peters, J. 2007c. Guided C^2 spline surfaces with V-shaped tessellation. In *Mathematics of Surfaces XII (Proceedings of 12th IMA International Conference)*. Vol. 4647. Springer Berlin / Heidelberg, London, UK, 233–244.

Karčiauskas, K. and Peters, J. 2007d. On the curvature of guided surfaces. Tech. rep., University of Florida CISE, REP-2007-430, Gainesville, FL, USA.

Karčiauskas, K. and Peters, J. 2008. On the curvature of guided surfaces. *Computer Aided Geometric Design 25,* 2, 69–79.

Kobbelt, L. 1996. Interpolatory subdivision on open quadrilateral nets with arbitrary topology. *Computer Graphics Forum (Proceedings of Eurographics) 15,* 3, 409–420.

Kobbelt, L. 2000. $\sqrt{3}$-subdivision. In *SIGGRAPH '00: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 103–112.

Levin, A. 2006. Modified subdivision surfaces with continuous curvature. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*. ACM Press, New York, NY, USA, 1035–1040.

Loop, C. 2002a. Bounded curvature triangle mesh subdivision with the convex hull property. *The Visual Computer 18,* 5–6 (Aug), 316–325.

Loop, C. 2002b. Smooth ternary subdivision of triangle meshes. In *Curve and Surface Fitting*. Vol. 10(6). Nashboro Press, Saint-Malo, 3–6.

Loop, C. T. 1987. Smooth subdivision surfaces based on triangles. Master's Thesis, Department of Mathematics, University of Utah.

Loop, C. T. and Schaefer, S. 2008. G^2 tensor product splines over extraordinary vertices. *Computer Graphics Forum (Proceedings of 2008 Symposium on Geometry Processing) 27,* 5, 1373–1382.

Lutterkort, D. and Peters, J. 2001. Tight linear envelopes for splines. *Numerische Mathematik 89,* 4 (Oct), 735–748.

Morin, G., Warren, J. D., and Weimer, H. 2001. A subdivision scheme for surfaces of revolution. *Computer Aided Geometric Design 18,* 5, 483–502.

Myles, A., Karčiauskas, K., and Peters, J. 2008. Pairs of bi-cubic surface constructions supporting polar connectivity. *Computer Aided Geometric Design 25,* 8, 621–630.

Peters, J. 2000. Patching Catmull-Clark meshes. In *SIGGRAPH '00: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 255–258.

Peters, J. 2002. C^2 free-form surfaces of degree (3,5). *Computer Aided Geometric Design 19,* 2, 113–126.

Peters, J. and Reif, U. 1997. The simplest subdivision scheme for smoothing polyhedra. *ACM Transactions on Graphics 16,* 4, 420–431.

Peters, J. and Reif, U. 2004. Shape characterization of subdivision surfaces: basic principles. *Computer Aided Geometric Design 21,* 6, 585–599.

Peters, J. and Reif, U. 2008. *Subdivision Surfaces*. Geometry and Computing, Vol. 3. Springer-Verlag New York, Inc., New York, NY, USA.

Peters, J. and Shiue, L. 2004. Combining 4- and 3-direction subdivision. *ACM Transactions on Graphics 23,* 4, 980–1003.

Peters, J. and Umlauf, G. 2000. Gaussian and mean curvature of subdivision surfaces. In *Proceedings of the 9th IMA Conference on the Mathematics of Surfaces*. Springer-Verlag, London, UK, 59–69.

Prautzsch, H. 1997. Freeform splines. *Computer Aided Geometric Design 14,* 3, 201–206.

Prautzsch, H., Boehm, W., and Paluszny, M. 2002. *Bezier and B-Spline Techniques*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Prautzsch, H. and Reif, U. 1999a. Degree estimates for C^k-piecewise polynomial subdivision surfaces. *Advances in Computational Mathematics 10,* 2, 209–217.

Prautzsch, H. and Reif, U. 1999b. Necessary conditions for subdivision surfaces. *Advances in Computational Mathematics 10*, 209–217.

Prautzsch, H. and Umlauf, G. 1998. A G^2-subdivision algorithm. In *Geometric Modelling, Dagstuhl, Germany, 1996*. Springer-Verlag, London, UK, 217–224.

Prautzsch, H. and Umlauf, G. 2000. A G^1 and G^2 subdivision scheme for triangular nets. *International Journal of Shape Modeling 6*, 21–35.

Reif, U. 1995. A unified approach to subdivision algorithms near extraordinary vertices. *Computer Aided Geometric Design 12,* 2, 153–174.

Reif, U. 1998. TURBS—topologically unrestricted rational B-splines. *Constructive Approximation. An International Journal for Approximations and Expansions 14,* 1, 57–77.

Reif, U. and Peters, J. 2005. Structural analysis of subdivision surfaces – a summary. In *Topics in Multivariate Approximation and Interpolation*, K. J. et al., Ed. Vol. 12. Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, 149–190.

Sabin, M. A. 1991. Cubic recursive division with bounded curvature. *Curves and Surfaces*, 411–414.

Sabin, M. A., Dodgson, N. W., Hassan, M. F., and Ivrissimtzis, I. P. 2003. Curvature behaviours at extraordinary points of subdivision surfaces. *Computer Aided Design 35*, 1047–1051.

Stam, J. 1998. Exact evaluation of Catmull-Clark subdivision surfaces at arbitrary parameter values. In *SIGGRAPH '98: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*. ACM, New York, NY, USA, 395–404.

Stam, J. and Loop, C. T. 2003. Quad/triangle subdivision. *Computer Graphics Forum 22,* 1, 79–86.

Umlauf, G. 2005. Analysis and tuning of subdivision algorithms. In *SCCG '05: Proceedings of the 21st Spring Conference on Computer Graphics*. ACM, New York, NY, USA, 33–40.

Velho, L. and Zorin, D. 2001. 4–8 subdivision. *Computer-Aided Geometric Design 18,* 5, 397–427. Special Issue on Subdivision Techniques.

Ying, L. and Zorin, D. 2004. A simple manifold-based construction of surfaces of arbitrary smoothness. *ACM Transactions on Graphics 23,* 3 (Aug.), 271–275.

Zorin, D. 2000. Smoothness of stationary subdivision on irregular meshes. *Constructive Approximation 16,* 3, 359–397.

Zorin, D. 2006. Constructing curvature-continuous surfaces by blending. In *SGP '06: Proceedings of the Fourth Eurographics Symposium on Geometry Processing*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 31–40.

BIOGRAPHICAL SKETCH

Ashish Myles was born in Varanasi, India. He was awarded the Bachelor of Science degree in computer science and engineering from the University of Florida in 2002. He received his master's degree in computer science in 2004 also at the University of Florida, and continued on for his Ph.D., specializing in graphics and smooth-surface geometry.