

NURBS with Extraordinary Points: High-degree, Non-uniform, Rational Subdivision Schemes

Thomas J. Cashman
University of Cambridge

Ursula H. Augsdörfer
University of Cambridge

Neil A. Dodgson
University of Cambridge

Malcolm A. Sabin
Numerical Geometry Ltd

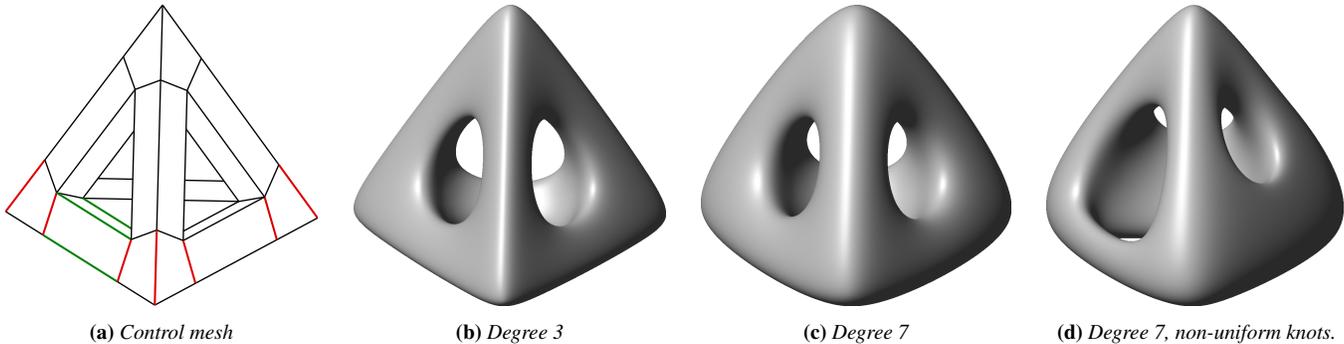


Figure 1: The same control mesh subdivided in three different configurations. In regular regions, the degree 3 surface (b) is C^2 , and the degree 7 surfaces (c) and (d) are C^6 . No previous subdivision scheme can generate the surface (d). The knot intervals modified to give (d) are shown in (a); the red interval is ten times greater than the unmarked intervals, and the green interval is four times greater. Comparing (d) with (c), note that in this case the non-uniform intervals change the whole surface, because the influence of a knot interval grows with degree.

Abstract

We present a subdivision framework that adds extraordinary vertices to NURBS of arbitrarily high degree. The surfaces can represent any odd degree NURBS patch exactly. Our rules handle non-uniform knot vectors, and are not restricted to midpoint knot insertion. In the absence of multiple knots at extraordinary points, the limit surfaces have bounded curvature.

CR Categories: I.3.5 [Computer Graphics]: Comp. Geom. and Object Modeling—Curve, surface, solid, and object representations

1 Introduction

Designers of freeform surfaces used to have to operate within the constraints imposed by NURBS [Farin 2001]. NURBS have a rigid rectangular control grid, so surfaces usually had to be represented as a collection of trimmed patches and continuity across the patch seams had to be manually enforced.

Subdivision surfaces removed these constraints by allowing designers to introduce extraordinary points into an otherwise regular control mesh. The resulting flexibility led to widespread adoption for applications such as character animation [DeRose et al. 1998], but NURBS are still an important industry standard in fields like

Computer-Aided Design. In addition to bicubic patches, CAD models sometimes use the additional continuity available from biquintic, occasionally bisepctic, and very rarely higher degree B-splines. High-class surfaces also use non-uniform parametrizations for handling boundaries, selectively reducing continuity, and integrating surface features at varying scales. No subdivision scheme, however, has been able to provide the full complement of NURBS features alongside the flexibility of extraordinary points.

We present a new subdivision framework that is a true superset of NURBS at odd degrees. We provide rules that give a continuous tangent plane and bounded curvature at the extraordinary points. Any existing NURBS patch can be represented exactly, and the regular regions of the resulting surfaces are also exact NURBS, so surfaces can be exported as a collection of NURBS to within any given deviation tolerance.

2 Related work

NURBS can be non-uniform and are well defined for arbitrarily high degree. Our subdivision scheme, as a superset of NURBS, must also hold these two properties (see Figure 1). Our method builds on previous schemes that address these challenges separately.

Interest in high-degree subdivision schemes initially centered around the search for surfaces with continuous non-zero curvature at extraordinary points. Reif [1996] showed that a stationary C^2 scheme would have to use at least bisextic patches, and both Prautzsch [1997] and Reif [1998] presented constructions which achieve this lower bound. The resulting surfaces are C^k at extraordinary points, for arbitrary k , and use B-spline patches with degree which increases rapidly with k .

Instead of specifying a continuity class (and letting degree adapt as necessary), Prautzsch [1998] also described a ‘midpoint’ scheme which can generate B-spline patches of a specified degree. The rules for this scheme generalize the averaging algorithm described by Lane and Riesenfeld [1980] for uniform B-spline subdivision.

By using a sequence of smoothing stages, the scheme avoids the problems arising from the widening influence of a point as degree increases. Warren and Weimer [2001] developed these ideas along with several other researchers; Zorin and Schröder [2001] showed that the resulting surfaces are C1 at extraordinary points (although the curvature diverges), and Stam [2001] described a variant where the topology of the mesh is invariant under smoothing.

Around the same time, subdivision surfaces were developed which generalize non-uniform (but fixed degree) B-spline surfaces. The tensor product construction for NURBS means that the same knot spacing must be shared by an entire “row” or “column” of edges, but Sederberg et al. [1998] presented surfaces which relax this property. Their scheme allows a separate knot spacing to be attached to every edge in a control mesh. Müller et al. [2006] developed a scheme which could additionally provide the position of a given point on the limit surface. These surfaces were also combined with T-splines [Sederberg et al. 2003], although with the restriction that opposite edges of a quadrilateral face must share the same knot spacing. This brought the surfaces into equivalence with bicubic NURBS, in the absence of extraordinary points and “T-junctions”.

There are therefore previous schemes that are true supersets of bicubic NURBS, and previous schemes of arbitrary degree, but no previous scheme that combines these – a superset of NURBS at arbitrary degree.

3 Overview

Like Prautzsch’s midpoint scheme, our method handles high degrees by breaking each subdivision step into one “refine” and several “smoothing” stages. We need to handle non-uniform knot vectors, however, and are therefore unable to use the Lane and Riesenfeld [1980] algorithm. Instead, we generalize a non-uniform analogue, which we review in Section 4.

We make the same restriction as Sederberg et al. [2003]: that each quadrilateral face must have equal knot spacing on opposite edges (as is the case with NURBS). We also restrict the control mesh to contain only quadrilateral faces (again, as with NURBS), but our rules must still allow for extraordinary vertices. We discuss the generalization that makes this possible in Section 5.

The resulting subdivision framework achieves our objective of adding extraordinary points to NURBS of arbitrary odd degree. Our method allows each knot interval to be subdivided at any position or not at all. In extraordinary regions, changing the sequence of knot insertions can change the limit surface, so it is important to resolve which knots we should insert, and where. We address this in Section 6, with an algorithm that first subdivides only large knot intervals until all the intervals are approximately the same size. This avoids problems which occur around extraordinary points if there is a large disparity between neighboring knot intervals. We use further subdivision steps to create a region of uniformity around extraordinary points, so that continuity and curvature variation in these regions is determined by uniform, rather than non-uniform, rules.

In Section 7, we use eigenanalysis to tune the uniform rules so that curvature is stable under the subdivision process, instead of increasing to infinity or collapsing to zero. The same bounded curvature conditions mean that extraordinary regions can hold both elliptic and hyperbolic shapes in the limit surface, as neither shape dominates the other. It proves more difficult to satisfy the bounded curvature conditions around vertices of valency 3; we found it necessary to introduce an additional smoothing stage to handle this case. With this modification, our method has bounded curvature for all valencies at arbitrarily high degrees, and maintains the convex hull property (by taking only convex combinations) up to degree 13.

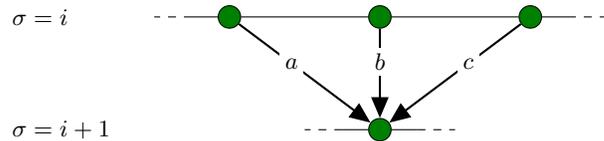


Figure 2: A smoothing stage on a single control point in the univariate knot insertion algorithm, where $a + b + c = 1$. Contributions come only from the point and its two immediate neighbors.

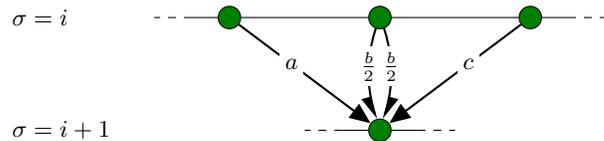


Figure 3: We can consider the contributions from each side of a point separately, by using only half weights for each of the contributions from the predecessor point.

4 Non-uniform, high-degree curves

As our method is built on NURBS, the subdivision rules for regular regions, whatever the degree, are derived from B-spline knot insertion. We can find these rules using a number of different approaches: two examples are the method presented by Boehm [1980] and the Oslo algorithm [Cohen et al. 1980]. All knot insertion algorithms give the same result but as degree increases, each point has a widening influence on points in the subdivided control polygon. For a generalization to surfaces containing extraordinary points, this causes problems at high degrees: both because of a drop in efficiency, and because a subdivision step must account for any possible configuration of extraordinary points in the control mesh. The number of cases to consider grows exponentially with the influence of a point, which grows quadratically with degree. Therefore using existing knot insertion rules directly is not a feasible option for high degrees.

Instead we need a knot insertion algorithm that uses a series of local computations which overlap, in the same way as the Lane and Riesenfeld [1980] algorithm. No matter how high the degree, the number of cases to consider is then constant. Schaefer and Goldman [2009] describe a univariate non-uniform algorithm which works in this way, but it is difficult to generalize to surfaces because of asymmetry. We developed a symmetric approach [Cashman et al. 2009], which we use here.

This univariate algorithm breaks a subdivision step into a refine stage followed by $\lfloor d/2 \rfloor$ smoothing stages, where d , which is odd, is the degree. We use σ to denote the current stage. At each smoothing stage we consider each point in turn, with the two knots which are a distance σ away in the subdivided knot vector on either side. If these knots need to be inserted, then the algorithm uses stage σ to introduce them into the *polar form* (blossom) [Ramshaw 1989] of the relevant point.

This process inserts 0, 1, or 2 knots into the polar form of each point at each stage. Cashman et al. [2009] give the required affine combinations and implementation details for the curve case. The important observation is that each affine combination uses only direct neighbors; we can therefore visualize the effect of a smoothing stage on a single point as shown in Figure 2, although either (or both) of the weights a and c may be 0. Figure 3 shows that we can consider the same smoothing operation one edge at a time, which forms the foundation of our method for the surface case.

5 Non-uniform, high-degree surfaces

To generalize the curve case to surfaces with extraordinary points, we calculate the affine combinations of refine and smooth stages *one face at a time*. The weights used in the affine combinations are defined using the tensor product of univariate knot insertion when considered *one edge at a time* (as shown in Figure 3). For a NURBS control mesh, each smoothing stage calculates weights in the four faces surrounding a vertex (Figure 4). These weights combine to give the tensor product of a smoothing stage in the form shown in Figure 2, as required for NURBS knot insertion.

5.1 Introducing extraordinary points

Like Sederberg et al. [1998], we associate a knot spacing with each edge of a control mesh. However, we require:

- that all faces are quadrilateral, and
- that knot spacings on opposite edges of a face are equal.

The latter is the same restriction that Sederberg et al. [2003] imposed on NURSS in order to incorporate T-junctions. The result is that knot spacings are defined for a whole strip of quadrilateral faces rather than for a single edge. Both conditions, of course, are properties of NURBS control meshes. In a mesh where these conditions hold, each face has two associated strips of quadrilateral faces, one in each of the two directions (Figure 5). We use these strips to build local knot vectors for each face. In a mesh containing extraordinary points, a smoothing stage may use different knot vectors for each face surrounding a vertex.

The refine stage is also computed one face at a time. Here we introduce vertices on existing edges and within existing faces. In the face case, the new vertex position is given by the tensor product of univariate refinement in the two directions. In the edge case, the two univariate contributions are split into four: two on each side of the edge. Cashman et al. [2009] give the univariate rules and Figure 6 shows an example of how these are applied to a bivariate case.

Section 7 describes our modifications for bounded curvature, which multiply the weight of each contribution from an extraordinary vertex by an additional value. There is a separate multiplier for contributions from an extraordinary vertex to its successor, α_n^d , to adjacent vertices, β_n^d , and to vertices across a diagonal, γ_n^d . The same value is used in every refine and smoothing stage, and is determined by the valency of the extraordinary point and the degree of the surface. Valency 3 is difficult (see Section 7.2) and requires an extra smoothing step to achieve bounded curvature while maintaining the convex hull property.

To maintain affine invariance, the weights in every affine combination must sum to 1. We therefore normalize each affine combination by dividing by the sum of contributing weights. This is necessary at extraordinary points, since a vertex of valency n receives $4n$ contributions instead of 16. The multipliers we use for bounded curvature make it necessary to normalize every affine combination that has a contribution from an extraordinary point. Normalization is sometimes necessary along the rays which emanate from extraordinary points, because the two sides of the ray can use different knot vectors. It is therefore easiest to normalize every affine combination, regardless of position in the mesh. This also handles faces with more than one extraordinary corner, without special cases.

This strategy allows us to create non-uniform subdivision surfaces at arbitrarily high degrees. To create a true superset of NURBS, all that remains is to make the surfaces rational, which is straightforward [Sederberg et al. 1998]: rational control points are projected into \mathbb{R}^4 , subdivided, and then projected back into \mathbb{R}^3 .

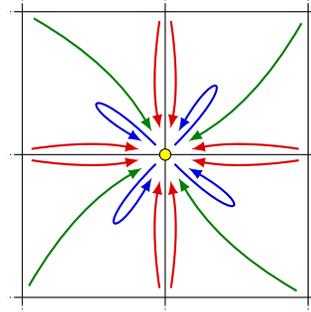


Figure 4: In regular regions, a smoothing stage computes contributions to a point \bullet from each of the surrounding four faces. The weights in each face are taken from the tensor product of Figure 3; the pairs of contributions shown in red define the influence of edge-connected vertices and the four contributions shown in blue combine to give the influence of a point on its successor. In a NURBS mesh, these 16 weights sum to 1, and compute the same affine combination as the tensor product of Figure 2.

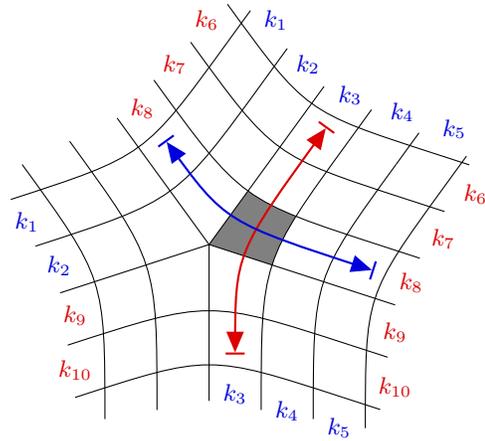


Figure 5: Every face has a local knot vector in two directions, which we can construct by following a strip of quadrilateral faces. For the shaded face, knot spacings in the blue direction are $\{k_1, k_2, k_3, k_4, k_5\}$ and in the red direction are $\{k_6, k_7, k_8, k_9, k_{10}\}$. This example collects knot vectors at the length required for subdivision at degree 5.

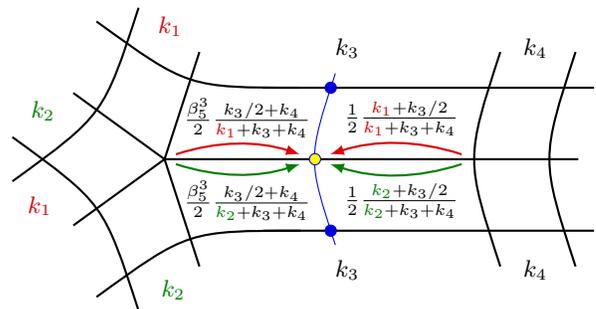


Figure 6: An example refine stage for degree 3, where the interval k_3 has a knot inserted at its midpoint. Other intervals are unmodified. The knot insertion algorithm therefore introduces points \bullet , \bullet along the subdivided interval, and we show the unnormalized weights used to form the central point \bullet . The fractions that involve k_i are from the curve case, the factors of one half are because we treat each side of each edge separately, and the multiplier, β_3^3 , is required to get bounded curvature in the bivariate case, as explained in Section 7.

6 Knot insertion strategy

We have now achieved our objective: NURBS of odd degree with extraordinary points. However, extraordinary points make it possible for a large knot interval, adjacent to a high-valency point, to be invisible to faces “hidden behind” the point. We find that this can cause folds in the limit surface if knots are inserted naïvely. Of course, in regular regions, as with NURBS, the sequence of knot insertions does not alter the limit surface. The same property does not apply around extraordinary points, so the sequence in which knots are inserted is important. In this section we describe our method for choosing which knots to insert, and where to insert them.

Our univariate algorithm [Cashman et al. 2009] is not constrained to subdivide every interval, so we can regularize knot spacings by subdividing large intervals first. Our regularization condition is that no knot interval is twice (or more) as large as any other. Given the complete set of knot intervals $\mathbf{k} = \{k_i\}$, we establish the minimum non-zero interval $\kappa = \min(\{k_i : k_i > 0\})$. If $\max(\mathbf{k}) \geq 2\kappa$, we use a limited subdivision step that inserts knots only into intervals which are 2κ or greater. We subdivide each interval at its midpoint, so a total of $\lfloor \log_2(\max(\mathbf{k})/\kappa) \rfloor$ such steps is required to achieve our regularization condition.

This strategy avoids undesirable folds in the surface at the scale of the first subdivision step. Once $\max(\mathbf{k}) < 2\kappa$, we address the behavior of the extraordinary points at smaller scales. To make guarantees on the behavior of these extraordinary regions, we use further subdivision steps to insert knots uniformly around extraordinary points, while subdividing other intervals at their midpoint. Doing so creates a uniform region with a stationary subdivision matrix. We are therefore able to apply eigenanalysis to guarantee C1 continuity (see the Appendix).

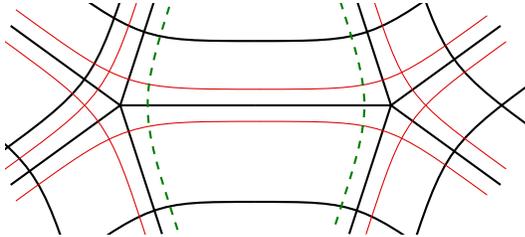


Figure 7: Two subdivision steps may be required to insert uniform knots around extraordinary points. In this figure, knot interval size is symbolized geometrically; black edges indicate original knots, and we insert the red knots in the next subdivision step. Note that the uniform spacing for the extraordinary point on the right is influenced by the intervals surrounding the point on the left. There is also a conflict in the central knot interval, as the knots shown using dashed green lines cannot both be inserted in the first subdivision step.

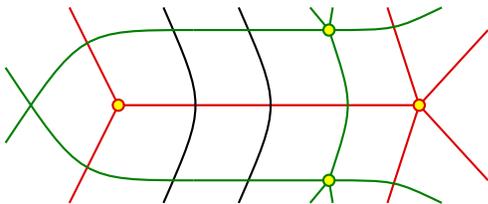


Figure 8: Our knot insertion strategy groups extraordinary points into equivalence classes based on whether they share an emanating ray. For this example, the two extraordinary points marked \bullet (with emanating rays drawn in red) are in the same equivalence class, as are the two points marked \bullet (with emanating rays drawn in green).

We need at most two steps before every extraordinary point is surrounded by one layer of evenly spaced knots. Consider the simplest case: a single extraordinary point of valency n surrounded by regular mesh. Let the knot intervals surrounding this point be k_1 to k_n . We insert a knot into each interval k_i , creating uniform spacing at a distance $\min(\{k_1, \dots, k_n\})/2$ away from the extraordinary point.

In the general case, we have multiple extraordinary points in an arbitrary configuration. Figure 7 shows that the uniform knot insertion for one extraordinary point may impact on another. In fact, extraordinary points with emanating rays which are shared over any number of regular points must insert knots at the same distance away, in order to achieve uniform knot spacing (Figure 8). We can define a set of equivalence classes for extraordinary points connected in this way, and calculate the membership of the classes in a single pass through points in the mesh. To achieve this, our implementation uses knot interval objects that hold a reference to an adjacent extraordinary point, if there is one, on either side. A disjoint set data structure [Galil and Italiano 1991], with associated “find” and “union” operations, is then used to join any classes which appear on the same side of a knot interval. The equivalence classes are unaffected by subdivision, so they need only be calculated once, and this can be at the coarsest level.

Once calculated, the equivalence classes can be used to insert knots at a distance which achieves uniform spacing for all members of the class. However, Figure 7 also shows that two or more extraordinary points, on either side of a knot interval, might make conflicting requests for where a knot should be inserted. In this case, we use the first subdivision step to insert a knot between the two positions. As the knot interval is then split, there can be no conflict in the second subdivision step.

Uniform knot spacing around extraordinary points is thus guaranteed in at most two steps. In summary, our algorithm for establishing uniform spacing is:

- for each equivalence class, establish the minimum from the knot intervals that surround extraordinary points within it
- for each extraordinary point, request to insert knots into surrounding intervals at half of the minimum interval for the class
- for each knot interval, subdivide at the midpoint, or at the mean of the requested positions, if there are insertion requests.

Note that the method described here cannot apply if there is a multiple knot (i.e. a zero knot interval) adjacent to an extraordinary point. In this case, we can still proceed with subdivision but we are unable to create uniform knot spacing, which means that we cannot *prove* C1 continuity in the presence of zero knot intervals.

Where uniform spacing is achievable, additional subdivision steps insert knots into the centers of the surrounding intervals. This creates a growing region of uniformity around each extraordinary point, so the behavior of the limit surface at these points is determined by uniform rules. Therefore Section 7 (which modifies the uniform case for bounded curvature) and the Appendix (which discusses our proof that the uniform rules create C1 surfaces) will apply to any non-uniform configuration with non-zero knot intervals adjacent to extraordinary points.

7 Bounded curvature

Without multiple knots, degree d B-splines (including the regular regions of our scheme) have $d - 1$ continuous derivatives. At extraordinary points, however, it has proved difficult for stationary subdivision to provide anything more than continuity of the first derivative. There are a range of possible behaviors for the second

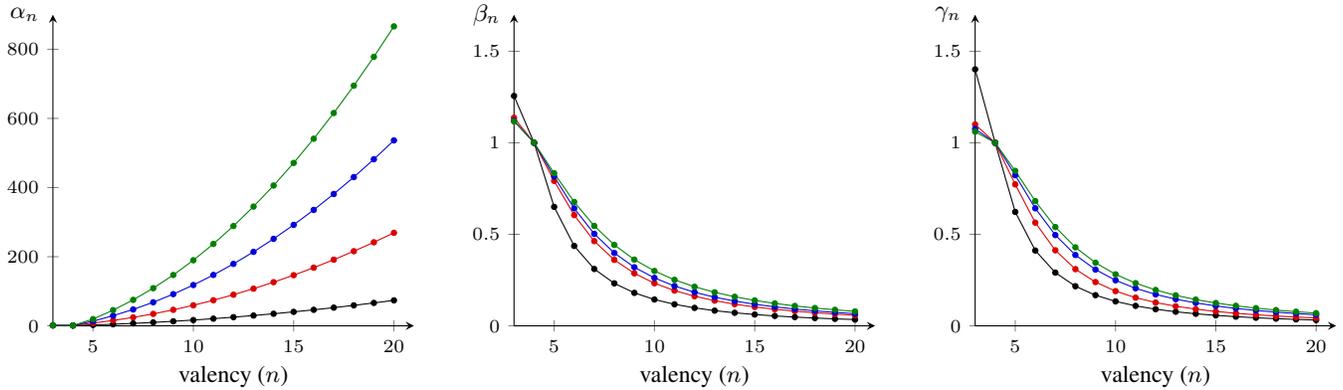


Figure 9: Bounded curvature solutions for α_n^d , β_n^d and γ_n^d . Valency ranges from 3 to 20. The multipliers are plotted for degree, d , equal to 3 —●—, 5 —●—, 7 —●— and 9 —●—. Note that $\alpha_4^d = \beta_4^d = \gamma_4^d = 1$ by definition, as the regular case is curvature continuous and hence already satisfies the bounded curvature equations $\lambda^2 = \mu_0 = \mu_2$.

n	α_n^3	α_n^5	α_n^7	α_n^9	n	β_n^3	β_n^5	β_n^7	β_n^9	n	γ_n^3	γ_n^5	γ_n^7	γ_n^9
3	1.3333	1.3333	1.3333	1.3333	3	1.2560	1.1370	1.1204	1.1172	3	1.4012	1.1003	1.0758	1.0608
5	2.6079	8.0641	13.574	19.510	5	0.6499	0.7919	0.8152	0.8339	5	0.6224	0.7727	0.8228	0.8466
6	4.6412	15.560	29.347	45.446	6	0.4364	0.6048	0.6417	0.6770	6	0.4115	0.5638	0.6420	0.6813
8	9.9085	34.571	68.153	108.89	8	0.2321	0.3604	0.3979	0.4421	8	0.2164	0.3098	0.3870	0.4289
20	73.691	269.09	536.37	866.17	20	0.0343	0.0578	0.0664	0.0790	20	0.0317	0.0430	0.0605	0.0695

Table 1: α_n^d , β_n^d and γ_n^d at a selection of valencies and with d ranging from 3 to 9. See auxiliary material for a full table.

derivative [Sabin et al. 2003], which we can analyze using an eigen-decomposition of the subdivision matrix [Peters and Reif 2008]. We aim for bounded curvature, which we believe gives the best behavior of the limit surface. We use λ to denote the subdominant eigenvalue, and write μ_0 and μ_2 for the eigenvalues of the components which generalize an elliptic and a hyperbolic paraboloid, respectively. As the subdivision rules we consider are rotationally symmetric, the eigenvalues λ and μ_2 are always double.

Reif [1996] shows that no stationary subdivision scheme based on bicubic pieces can have non-zero curvature continuity at extraordinary points. It is possible, however, to satisfy bounded curvature, where $\lambda^2 = \mu_0 = \mu_2$. This is a necessary, but not sufficient, condition for C2 continuity, and preserves curvatures in both of the quadratic eigenvectors through subdivision. Therefore the extraordinary region is not dominated by either the elliptic or hyperbolic component, and curvature at the extraordinary point is bounded by the curvatures appearing in the spline rings of those components [Peters and Reif 2008]. In contrast, the Catmull-Clark scheme [Catmull and Clark 1978] has $\lambda^2 < \mu_0 < \mu_2$ at valencies greater than four, leading to a surface which is always hyperbolic, irrespective of the control mesh shape [Karciauskas et al. 2004].

Several authors have modified stationary subdivision schemes to improve curvature in extraordinary regions. Sabin [1991] presents a variant of Catmull-Clark with bounded curvature. Barthe and Kobbelt [2004] tune extraordinary point rules for a number of desirable properties, including bounded curvature, and both Holt [1996] and Loop [2002] present bounded curvature schemes for triangle meshes. Augsdörfer et al. [2006] tune schemes for bounded curvature while seeking to minimize the range of the bounds, and Ginkel and Umlauf [2006] try to eliminate cases where the bounded range includes both positive and negative curvatures.

If the subdivision matrix is not required to be stationary, then curvature continuity is easier to achieve. Levin [2006], for example, uses

non-stationary rules that result in a local blend between the limit surface and a C2 polynomial. It would be possible to apply the same principle to our scheme, once uniform knot spacing has been created around extraordinary points. However we have chosen to maintain the simplicity of stationary rules for uniform configurations. We now show how to modify our rules to give bounded curvature at arbitrarily high degree.

7.1 Valency greater than regular

For uniform knot vectors at each degree and valency, we compute the Discrete Fourier Transform (DFT) of the subdivision matrix for the rules described in Section 5. We want to modify the rules for bounded curvature, but the size of the subdivision matrix increases with degree. It therefore becomes impractical to find an exact solution in symbolic form, even using the DFT and considering each block separately. Instead we use a nonlinear solver on floating point values. We ensure that λ^2 differs from μ_0 and μ_2 by no more than 10^{-12} (with subdivision matrices calculated to double precision, and with floating-point eigenvalue routines). Note that this process is offline, and has no impact on the speed of an implementation.

We modify our rules using mask tuning [Augsdörfer et al. 2006], adapted for the setting where a subdivision step is factorized into separate stages. The idea is to multiply the weight of *any contribution from an extraordinary point* by an additional value. We use the same multipliers for the refine and smoothing stages. An extraordinary vertex can make contributions to vertices in three different positions, each of which uses a separate multiplier:

- α_n^d , for the new extraordinary vertex,
- β_n^d , for edge-connected vertices,
- γ_n^d , for vertices diagonally opposite the extraordinary vertex in adjacent faces.

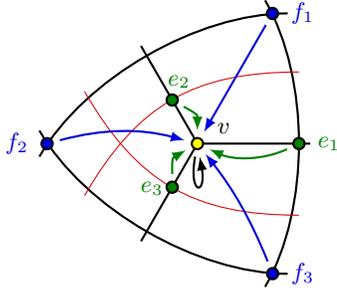


Figure 10: Final affine combination used to position 3-valent vertices for bounded curvature. Red lines indicate knots which have been inserted in this subdivision step (as an illustration, we show a case where no knot has been inserted into the interval on the right). The extraordinary point, in the center, takes a new position calculated from the existing position v (●), the points e_i (●) which are edge-connected at the end of the subdivision step, and the points f_i (●) which are in the positions which were face-connected at the start of the subdivision step.

δ_3	δ_5	δ_7	δ_9	δ_{11}	δ_{13}
0.0013	0.0813	0.1915	0.3537	0.5732	0.8550

Table 2: Values for δ_d to achieve bounded curvature for valency 3.

```
// Initialization
foreach extraordinary point p do
  assign p to an equivalence class

foreach subdivision step do
  // Knot insertion strategy
  if max(k) ≥ 2κ then
    foreach knot interval k where k ≥ 2κ do
      subdivide k at its midpoint
  else
    foreach equivalence class e_i do
      κ_i ← min(knot intervals surrounding points in e_i)
    foreach extraordinary point p in equivalence class e_i do
      foreach knot interval surrounding p do
        make insertion request at a distance 1/2 κ_i from p
    foreach knot interval k do
      if k has insertion requests then
        subdivide k at mean of requested positions
      else
        subdivide k at its midpoint

  // Subdivide mesh based on inserted knots
  foreach quadrilateral face f with knot intervals k_i and k_j do
    if k_i and k_j both subdivided then
      compute refine stage for new point within f
    if k_i or k_j is subdivided then
      make contributions for new points on edge of f

  for σ ← 1 to ⌊degree/2⌋ do
    foreach quadrilateral face f do
      make contributions for smoothing stage σ within f

  foreach 3-valent point p do
    compute final position of p
```

Figure 11: Pseudocode for our subdivision algorithm

We find bounded curvature solutions for each d , the degree (which is always odd), and each n , the valency. In an implementation, these values are stored in a lookup table.

The multiplier α affects only the zero frequency Fourier block, so we can first solve $\lambda^2 = \mu_2$ using just β and γ . We use the second degree of freedom in this optimization to make the shape of the subdominant eigenvector as stable as possible. Taking these solutions into the zero frequency block, we are then able to use α to solve $\lambda^2 = \mu_0$. Figure 9 and Table 1 show the resulting solutions for degrees 3 to 9 and valencies up to 20. A full table is included as auxiliary material.

7.2 Valency 3

At valency 3, we can use the method described above to solve $\lambda^2 = \mu_2$ in exactly the same way. If we try to satisfy $\lambda^2 = \mu_0$, however, we find that the solutions for α_3^d are negative. The resulting surfaces could therefore lie outside the convex hull of the control points. Even worse, there are some degrees with no bounded curvature solution at all. We therefore need to treat the valency 3 case separately.

Instead of using a negative weight for the contribution from an extraordinary vertex to its successor, we can instead shrink μ_0 by increasing the number of points which influence the extraordinary vertex over the course of a subdivision step. This increase in support may be considered undesirable, but we consider convex combinations more important than a small increase in support.

Our modification introduces an additional, final smoothing stage, affecting only 3-valent vertices (Figure 10). We need to take a contribution from vertices that are sufficiently far away from the extraordinary point to shrink μ_0 at a rate of λ^2 . We also want to avoid complications arising from arbitrary connectivity. Our solution is therefore to take f_i , the vertices in the positions which were face-connected to the extraordinary point at the start of the subdivision step. In the uniform case, the f_i therefore lie at the corners of the two rings surrounding an extraordinary point. If a face has not been subdivided in one direction or both, however, they will be more closely connected. In Figure 10, for example, f_1 and f_3 are only a knight's move away from the extraordinary point. We also use e_i , the vertices connected to the extraordinary point by edges, and the position of the 3-valent extraordinary point, v . The position of the new point, \hat{v} , is given by

$$\hat{v} = \rho v + \sum_{i=1}^3 \frac{1}{3} (1 - \rho) (1 - \delta_d) e_i + \sum_{i=1}^3 \frac{1}{3} (1 - \rho) \delta_d f_i$$

This final smoothing stage allows us to solve $\lambda^2 = \mu_0$ using the same numerical solver, but varying δ_d instead of α_3^d (see Table 2). Here ρ is the product, over every stage, of the normalized weight in the contribution from the extraordinary point to its successor. We use ρ to retain the property that a point is only modified if knots are inserted nearby. If no knots are inserted into the domain of the extraordinary point's basis function, then $\rho = 1$ gives $\hat{v} = v$ as required.

Every affine combination in our univariate knot insertion algorithm uses non-negative weights. The same is therefore true of the tensor product, and of the modifications for bounded curvature, as the multipliers α , β and γ are also non-negative. Therefore $0 \leq \rho \leq 1$, as ρ is the product of non-negative weights, and if $0 \leq \delta_d \leq 1$, then our scheme holds the convex hull property.

In solving $\lambda^2 = \mu_0$, we find $0 \leq \delta_d \leq 1$ for the six degrees where d ranges from 3 to 13. The resulting values are given in Table 2. With the inclusion of an additional smoothing stage, α_3^d has relatively little effect, as the position of the extraordinary point is largely determined by the value of δ_d . However, we do want to make

Control mesh	Degree 3	Degree 5	Degree 7	Degree 9

Table 3: Surfaces with three different knot configurations (top to bottom) at degrees 3, 5, 7 and 9 (left to right). Each surface is drawn with reflection lines and is defined by the control mesh shown on the left. In each row, the knot intervals shown in red are five times larger than the others; the first row shows the uniform case. Note the widening influence of a knot interval as degree increases. The six surfaces which are both high-degree and non-uniform (bottom-right) cannot be generated by any previous subdivision scheme.

sure that $\delta_3 \geq 0$, and therefore we set $\alpha_3^d = \frac{4}{3}$ for all d as a useful simplification that achieves this. Given uniform knot spacing, this value scales contributions from 3-valent points to their successors so that the total unnormalized weight is the same as in the regular, 4-valent case.

In summary, we have now modified general degree, non-uniform subdivision schemes to have bounded curvature at all degrees and valencies. The schemes hold the convex hull property for a useful range of degrees from 3 to 13, which, at valency 3, requires a small increase in the support of surrounding basis functions. The complete algorithm is summarized as pseudocode in Figure 11.

8 Discussion

This work follows calls from the CAD industry for subdivision surfaces that do not enforce a uniform parametrization [Gonsor and Neamtu 2001]. Ma [2005] also suggests, as a research area that would allow subdivision to gain greater acceptance in CAD, “a further unified generalization ... [which] should cover all [that] we can do with NURBS.” We have achieved this for odd degree.

Figure 1 and Table 3 show examples at a range of degrees and different knot intervals. Karciauskas et al. [2004] note that bounded curvature schemes can suffer from a lack of fairness. This effect seems to be visible in the degree 3 surfaces of Table 3, since the reflection lines are not as smooth as a Catmull-Clark surface on the same data. However, we have recourse to higher degrees for providing additional smoothness, and the surfaces at degree 5 and higher seem fairer. In fact, we believe that even in the completely uniform case, the degree 5 surfaces are an interesting alternative

to Catmull-Clark. The computational cost is only slightly higher (being degree 5 rather than degree 3), and yet the resulting surfaces have bounded curvature without compromising global shape.

Around points with high valency, subdivision surfaces typically suffer from poor shape (Figure 12). Our method produces a much smoother shape, even at low degree, compared with schemes that have divergent curvature. There is also a noticeable improvement in visual smoothness by using a higher degree, although the surface remains formally only C1 at the extraordinary point.

Zorin and Schröder [2001] found that general degree surfaces which are generated by centroid averaging have curvatures which diverge faster, at extraordinary points, as degree increases. Our modifications for bounded curvature avoid this outcome, so there are fewer disadvantages to using higher degrees to obtain greater smoothness. The remaining disadvantages apply to B-splines in general: higher computational cost, increased shrinkage when comparing the limit surface to the control mesh, and control points which have a less local influence over the surface.

Our modifications for bounded curvature require λ to be less than $\frac{1}{2}$ for valency 3, and greater than $\frac{1}{2}$ for valency greater than 4 (see Figure 14). Barthe and Kobbelt [2004] explicitly try to avoid this outcome, because of the “polar artifact” which then appears when a subdivision surface is rendered using its control polygon after just a few subdivision steps. However, we are primarily interested in the behavior of the limit surface (made up of B-spline patches) rather than the control mesh, as it is the limit surface that is important for CAD applications.

Non-uniform parametrizations might be useful where subdivision surfaces are textured (for example, procedurally) using the piecewise

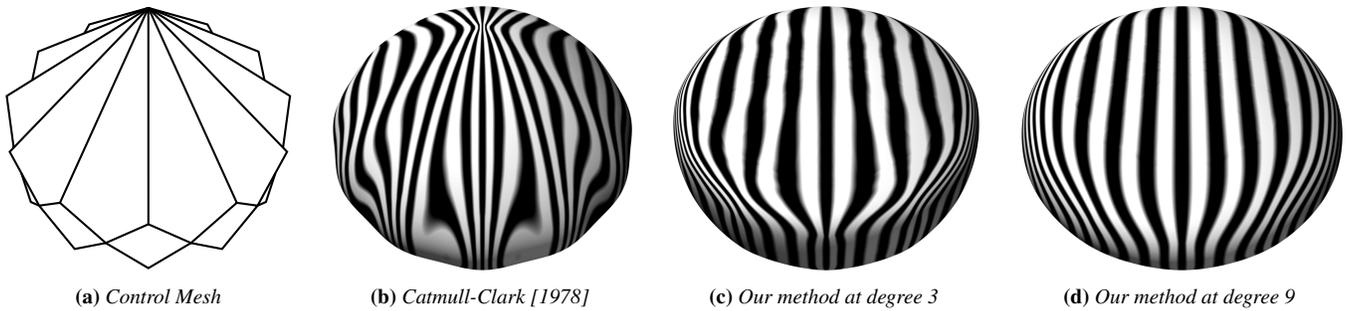


Figure 12: Reflection lines on surfaces defined by a control mesh with a 12-valent extraordinary point. Each surface is rendered with vertices projected to the limit surface; (c) and (d) use exact evaluation [Stam 1998] to sample the limit surface in the region of the 12-valent point.

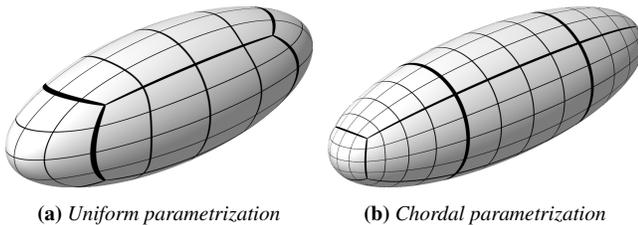


Figure 13: A cuboid subdivided at degree 3. The subdivision surfaces are shown with isoparameter lines. Setting knot intervals to chord length increases the regularity of the parametrization.

parametrization associated with the control mesh. Like NURBS, our scheme allows each face to be parametrized by a rectangle, rather than the unit square. Figure 13 shows that a chordal parametrization can help reduce texture distortion by increasing the regularity of this parametrization.

Another use for non-uniform parametrizations is to allow good control of the boundaries of subdivision surfaces. We have previously described how to handle boundaries in the curve case [Cashman et al. 2009] and this is easily transferred to the surface case. As a result, we can use multiple knots to give surfaces which meet a univariate B-spline curve at the boundary, whilst providing good control of both tangent and curvature. Existing boundary conditions, such as duplicating control points to gain a clamped surface, suffer from zero curvature at the boundary.

With the introduction of subdivision surfaces which have the full capabilities of NURBS, it is possible that end users need no longer be presented with mutually exclusive surface primitives for NURBS and Catmull-Clark subdivision surfaces. We believe that the surfaces presented in this paper could allow many applications to present a single, unified interface to users, allowing control which is familiar from NURBS (non-uniform parametrizations at arbitrary degree) alongside the flexibility of extraordinary points, which is familiar from subdivision.

There are two areas in particular where there may be better solutions than the ones we have presented: the affine combination used for bounded curvature at valency 3 (Section 7.2), and the method used for creating uniform knot spacing (Section 6). A natural development would be to make the condition $\max(k) < 2\kappa$ local rather than global. A knot interval would then only be considered too large if there was an interval of half the size sufficiently nearby (where this distance depends on degree). This modification would allow normal midpoint subdivision for knot intervals which increase

gradually across a surface, without compromising continuity around extraordinary points.

There might also be the potential for more extensive changes. We could allow faces to have a different knot spacing on opposite edges like Sederberg et al. [1998] and Müller et al. [2006]. This would involve a generalization of the knot vectors which are collected as shown in Figure 5. With this generalization in place, it might be possible to incorporate extraordinary faces as well as extraordinary vertices.

As we mentioned in Section 6, our method makes no guarantees on surface continuity if zero knot intervals lie adjacent to extraordinary points. Furthermore, around extraordinary regions, we are unable to insert an arbitrary knot while maintaining an invariant limit surface. Neither of these limitations affects compatibility with NURBS, but they allow room for further analysis and improvement.

Finally, the framework we describe here has a natural counterpart for even degrees, where knot spacings are specified across a strip of edges rather than a strip of faces. In future work, we hope to provide bounded curvature modifications for the even degree schemes.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful feedback. Ursula Augsdörfer and Malcolm Sabin were supported by EPSRC grant number EP/E025889/1.

References

- AUGSDÖRFER, U. H., DODGSON, N. A., AND SABIN, M. A. 2006. Tuning Subdivision by Minimising Gaussian Curvature Variation Near Extraordinary Vertices. *Comp. Graph. Forum* 25, 3, 263–272.
- AUGSDÖRFER, U. H., CASHMAN, T. J., DODGSON, N. A., AND SABIN, M. A. 2009. Numerical Checking of C^1 for Arbitrary Degree Quadrilateral Subdivision Schemes. In *13th IMA Conference on the Mathematics of Surfaces*, Springer. To appear.
- BARTHE, L., AND KOBBELT, L. 2004. Subdivision scheme tuning around extraordinary vertices. *CAGD* 21, 6, 561–583.
- BOEHM, W. 1980. Inserting new knots into B-spline curves. *Computer-Aided Design* 12, 4, 199–201.
- CASHMAN, T. J., DODGSON, N. A., AND SABIN, M. A. 2009. Selective knot insertion for symmetric, non-uniform refine and smooth B-spline subdivision. *CAGD* 26, 4, 472–479.

- CATMULL, E., AND CLARK, J. 1978. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-Aided Design* 10, 6, 350–355.
- COHEN, E., LYCHE, T., AND RIESENFELD, R. 1980. Discrete B-splines and Subdivision Techniques in Computer-Aided Geometric Design and Computer Graphics. *Computer Graphics and Image Processing* 14, 2, 87–111.
- DEROSE, T., KASS, M., AND TRUONG, T. 1998. Subdivision surfaces in character animation. In *Proc. SIGGRAPH 98*, 85–94.
- FARIN, G. 2001. *Curves and Surfaces for CAD: A Practical Guide*, 5th ed. Morgan Kaufmann.
- GALIL, Z., AND ITALIANO, G. 1991. Data structures and algorithms for disjoint set union problems. *ACM Computing Surveys* 23, 3, 319–344.
- GINKEL, I., AND UMLAUF, G. 2006. Loop subdivision with curvature control. In *Eurographics Symposium on Geom. Proc.*, Eurographics, K. Polthier and A. Sheffer, Eds., 163–171.
- GONSOR, D., AND NEAMTU, M. 2001. Subdivision Surfaces – Can they be Useful for Geometric Modeling Applications? Tech. Rep. 01-011, The Boeing Company.
- HOLT, F. 1996. Toward a curvature-continuous stationary subdivision algorithm. *Zeitschrift für angewandte Mathematik und Mechanik* 76, 423–424.
- KARCIAUSKAS, K., PETERS, J., AND REIF, U. 2004. Shape characterization of subdivision surfaces—case studies. *CAGD* 21, 6, 601–614.
- LANE, J. M., AND RIESENFELD, R. F. 1980. A Theoretical Development for the Computer Generation and Display of Piecewise Polynomial Surfaces. *IEEE Trans. PAMI* 2, 1, 35–46.
- LEVIN, A. 2006. Modified subdivision surfaces with continuous curvature. *ACM Trans. Graph.* 25, 3, 1035–1040.
- LOOP, C. 2002. Bounded curvature triangle mesh subdivision with the convex hull property. *The Visual Computer* 18, 316–325.
- MA, W. 2005. Subdivision surfaces for CAD—an overview. *Computer-Aided Design* 37, 7, 693–709.
- MÜLLER, K., REUSCHE, L., AND FELLNER, D. 2006. Extended subdivision surfaces: Building a bridge between NURBS and Catmull-Clark surfaces. *ACM Trans. Graph.* 25, 2, 268–292.
- PETERS, J., AND REIF, U. 2008. *Subdivision Surfaces*. Springer.
- PRAUTZSCH, H. 1997. Freeform splines. *CAGD* 14, 3, 201–206.
- PRAUTZSCH, H. 1998. Smoothness of subdivision surfaces at extraordinary points. *Adv. in Comp. Math.* 9, 3, 377–389.
- RAMSHAW, L. 1989. Blossoms are polar forms. *CAGD* 6, 4, 323–358.
- REIF, U. 1996. A Degree Estimate for Subdivision Surfaces of Higher Regularity. *Proc. Amer. Math. Soc.* 124, 7, 2167–2174.
- REIF, U. 1998. TURBS—Topologically Unrestricted Rational B-Splines. *Constructive Approximation* 14, 1, 57–77.
- SABIN, M. A., DODGSON, N. A., HASSAN, M. F., AND IVRISIMTZIS, I. P. 2003. Curvature behaviours at extraordinary points of subdivision surfaces. *Computer-Aided Design* 35, 11, 1047–1051.
- SABIN, M. 1991. Cubic recursive division with bounded curvature. In *Curves and surfaces*, Academic Press, 411–414.
- SCHAEFER, S., AND GOLDMAN, R. 2009. Non-uniform Subdivision for B-splines of Arbitrary Degree. *CAGD* 26, 1, 75–81.
- SEDERBERG, T. W., ZHENG, J., SEWELL, D., AND SABIN, M. 1998. Non-Uniform Recursive Subdivision Surfaces. In *Proc. SIGGRAPH 98*, 387–394.
- SEDERBERG, T. W., ZHENG, J., BAKENOV, A., AND NASRI, A. 2003. T-splines and T-NURCCs. *ACM Trans. Graph.* 22, 3, 477–484.
- STAM, J. 1998. Exact evaluation of Catmull-Clark subdivision surfaces at arbitrary parameter values. In *Proc. SIGGRAPH 98*, 395–404.
- STAM, J. 2001. On subdivision schemes generalizing uniform B-spline surfaces of arbitrary degree. *CAGD* 18, 5, 383–396.
- WARREN, J., AND WEIMER, H. 2001. *Subdivision Methods for Geometric Design*. Morgan Kaufmann.
- ZORIN, D., AND SCHRÖDER, P. 2001. A unified framework for primal/dual quadrilateral subdivision schemes. *CAGD* 18, 5, 429–454.

Appendix: C1 continuity

Our strategy for knot insertion (Section 6) creates a region of uniformity around every extraordinary point surrounded by non-zero knot intervals. Therefore the continuity of the limit surface is determined by a stationary subdivision matrix, and we can use eigenanalysis to prove that our method produces C1 surfaces.

The proof uses Theorem 5.25 from Peters and Reif [2008]. The subdivision matrices derived from our method satisfy the conditions of the theorem: they are flip-symmetric, shift-invariant, and the subdominant eigenvalues lie in the unit frequency Fourier block. It remains to show that the derivatives of the *characteristic ring* (which is made up of B-spline patches, of the degree of the scheme) do not overlap. Due to symmetry, we need only check that the derivatives in one direction of the first sector (drawn red in Figure 14) lie in the same quadrant [Peters and Reif 2008]. This is enough to prove that the characteristic ring is regular and injective, and thus that the schemes are C1.

Using the variation diminishing property of B-splines, we can prove this condition on the derivatives using the first differences of points in the subdominant eigenvector [Augsdörfer et al. 2009]. Where it is necessary to obtain a tighter bound, we can use subdivision steps to bring the first differences closer to the derivative. These subdivision steps require only straightforward uniform B-spline knot insertion. We have verified this condition, and hence proved C1 continuity, for degree from 3 to 19 and valency from 3 to 50. The trends suggest that no problem is likely to occur at higher valencies and degrees.

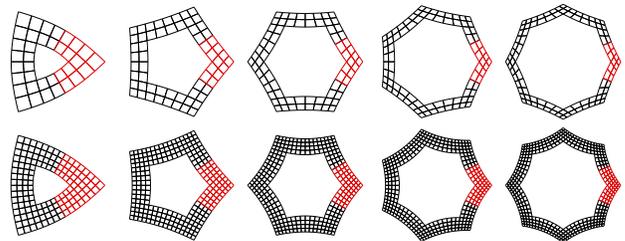


Figure 14: Characteristic rings [Peters and Reif 2008] at valencies 3 and 5 to 8 (left to right) and degrees 3 (top) and 7 (bottom). The first sector is drawn in red.