Multiresolution on Spherical Curves

Troy Alderson, Ali Mahdavi Amiri, Faramarz Samavat

Department of Computer Science, University of Calgary

Abstract

In this paper, we present an approximating multiresolution framework of arbitrary degree for curves on the surface of a sphere. Multiresolution by subdivision and reverse subdivision allows one to decrease and restore the resolution of a curve, and is typically defined by affine combinations of points in Euclidean space. While translating such combinations to spherical space is possible, ensuring perfect reconstruction of the curve remains challenging. Hence, current spherical multiresolution schemes tend to be interpolating or midpoint-interpolating, as achieving perfect reconstruction in these cases is more straightforward. We use a simple geometric construction for a non-interpolating and non-midpoint-interpolating multiresolution scheme on the sphere, which is made up of easily generalized components and based on a modified Lane-Riesenfeld algorithm.

Keywords: Subdivision, Reverse Subdivision, Multiresolution, Spherical Space

1. Introduction

The question of how to decrease the resolution of a curve and restore it to its original state is a well-studied subject in computer graphics, and falls under the purview of *multiresolution frameworks*. Applications include level-of-detail control, compression, and multiscale editing for curves. Such frameworks can be created using a combination of subdivision and reverse subdivision [1].

In Euclidean space, subdivision schemes are linear transformations that increase the resolution of a curve or surface, while reverse subdivision schemes are linear transformations that decrease the resolution. Many subdivision schemes are based on B-Spline basis functions, and converge to B-Spline curves or surfaces at the limit. Chaikin's corner-cutting scheme for curves [2] as well as the Catmull-Clark [3] scheme for surfaces are some well-known examples of B-Spline subdivision schemes for which reverse methods have been proposed. Both forward and reverse subdivision are often understood and implemented using affine combinations of points, specified by simple linear filters.

When combined into a multiresolution framework, a given vector of m fine points $f = [f_0 \dots f_{m-1}]^T$ can be decomposed to a vector of $n < m$ coarse points $c = [c_0 \dots c_{n-1}]^T$ and associated detail vectors (or wavelet coefficients) $d = [d_0 \dots d_{m-n-1}]^T$ [4, 5], then reconstructed using c and d. A notable property of such a framework is that the total number of coarse points and details is equal to the original number of points before decomposition. As a result, no additional information is needed to fully retrieve the high resolution data. Furthermore, these operations are both fast and efficient.

While well understood in 2D or 3D Euclidean space, achieving multiresolution via subdivision and reverse subdivision in other spaces is a challenging but fascinating topic of study. The sphere, for instance, is an elegant and important geometric domain, and of particular interest as an approximation of the shape of the Earth [6]. However, its surface forms a two-dimensional non-Euclidean space in which many traditional geometric intuitions do not apply. Curves in spherical space — analogous to curves in Euclidean space — are called spherical curves and are formed by an ordered set of points f_i on the sphere connected by geodesic lines (great circle arcs).

Our work focuses on decreasing and increasing/restoring the resolution of spherical curves (i.e. spherical multiresolution) based on B-Spline subdivision and reverse subdivision, with an intended application in vector data representation on the spherical surface of a Digital Earth [7, 8]. Geospatial vector data are often very large (consisting of thousands of points) and can benefit from multiscale representations due to their support for compression, progressive transmission over networks, level-of-detail control in visualization, and fast estimates for queries.

In general, the fundamental challenge in spherical multiresolution lies in translating affine combinations of points to spherical space in a manner that ensures the scheme is loss-less (i.e. perfect reconstruction of the original fine data f is achieved).

A straightforward solution is to project the points of the spherical curve to a Euclidean domain (e.g. using a spherical projection from the field of cartography), apply affine combinations in that domain, and project back to the sphere. Potential mappings include latitude/longitude or spherical coordinate conversion, which is a standard projection; Snyder projection [9], which is an equal area projection often encountered in Digital Earth frameworks; and the exponential map [10], which maps points to a local tangent plane. Unfortunately, as the spherical and Euclidean space are not isometric, this approach often introduces distortions into the resulting curves (see Figure 1).

A second approach is to generalize the affine combination $p = a_0q_0 + a_1q_1 + \cdots + a_{n-1}q_{n-1}$ ($p, q_i \in \mathbf{R}^3, a_i \in \mathbf{R}$) to spherical

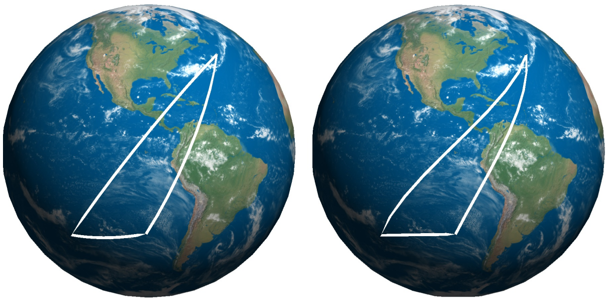


Figure 1: A spherical curve defined by three points is shown (on left). After mapping the points to latitude/longitude coordinates, drawing Euclidean lines between the resulting points, and mapping those lines back to the sphere, significant mapping distortions are revealed (on right).

space as the (local) solution to

$$\min_p \left\| \sum_{i=0}^{n-1} a_i \cdot \exp_p(q_i) \right\|,$$

as in [11], where $\exp_p(q_i)$ is the exponential map operator that maps q_i to a vector in the tangent space of p. However, due to the nature of non-Euclidean space, generalizing these combinations in this manner does not in general result in a scheme with perfect reconstruction. Hence, the work of [11] focuses on interpolating and midpoint-interpolating multiresolution schemes, for which perfect reconstruction can be guaranteed. As a non-interpolating and non-midpoint-interpolating (i.e. approximating) scheme, B-Spline multiresolution is non-trivial to translate to spherical space.

A third approach, as seen in [12, 13] and the one adopted in this paper, is to split the affine combinations into series of two-point interpolations. Such two-point interpolations are atomic operations in spherical space that are analogous to the simplest atomic operations used to create curves in Euclidean space, and can be computed efficiently using spherical linear interpolation (SLERP), defined by [14]

$$\mathrm{SLERP}(p, q, u) = \frac{\sin[(1-u)\theta]}{\sin(\theta)} p + \frac{\sin(u\theta)}{\sin(\theta)} q$$

(where θ is the angle between p and q). Unlike Euclidean space, in which any reformulation of an affine combination into two-point interpolations will have the same result, in spherical space different reformulations of the affine combination give different results. Hence, it is again difficult to ensure perfect reconstruction in this case.

We present in this paper a construction of a loss-less approximating multiresolution scheme in spherical space (inspired by Euclidean B-Spline multiresolution) made up of sequences of two-point interpolations (i.e. SLERP operations). This holds for all constituent operations of the multiresolution: subdivision, reverse subdivision, detail computation (i.e. decomposition), and detail restoration (i.e. reconstruction). The construction is inspired by the Lane-Riesenfeld subdivision algorithm for B-Spline subdivision of arbitrary degree (or smoothness) in Euclidean space [15], which uses two atomic operations: point duplication and midpoint finding. Although easily generalized to the sphere, the algorithm does not have a corresponding reverse subdivision or multiresolution algorithm due to the non-invertibility of the midpoint finding operation.

Our construction, which can reproduce at least some of the B-Spline subdivisions returned by the Lane-Riesenfeld algorithm, replaces pairs of midpoint-finding operations with discrete smoothing operators that have local inverses in Euclidean and spherical space. Detail vectors d_i are generalized to detail *rotations* in spherical space, and are easy to compute and restore during reconstruction.

Furthermore, our multiresolution scheme includes reverse subdivision, detail computation, and detail restoration constructions based on atomic operations; to our knowledge the first of their kind. We expect translations of this scheme to more general manifolds are possible as well, provided an operation analogous to SLERP is defined on the manifold.

The paper is organized as follows. In Section 2, we describe previous works that are related to this problem. A generalization of the Lane-Riesenfeld algorithm to spherical space from [13] is described in Section 3, followed by a generalization to spherical space of [16]'s modified Lane-Riesenfeld algorithm with invertible averaging step in Section 4. In Sections 5, 6, and 7, we present our spherical multiresolution scheme, with some comments on analysis in Section 8. Results and comparisons follow in Section 9.

2. Related Work

Curves that lie on surfaces (including spheres) have been the subject of much research [17, 7, 8]. Spherical curves are especially important, as the sphere is an important shape in Geomatics and GIS and serves as an important intermediate shape for applications such as parametrization and illumination [18, 19]. Spherical curves are particularly of interest within the Digital Earth framework [20, 21, 22, 6], which represents the Earth as a curved surface rather than as a flattened map.

Multiresolution for curves and surfaces is also a well studied subject [23, 24, 25]. One means of establishing a multiresolution framework is to combine subdivision and reverse subdivision, in which the former produces a more detailed object while the latter reduces the resolution [4, 5, 26]. The convergence and smoothness of the limit curve of a subdivision scheme can be analysed using the techniques in [27] for Euclidean space and [12, 28] for manifold surfaces. In a multiresolution framework based on subdivision and reverse subdivision, no details are lost and all information needed to reconstruct the curve occupies no more memory than the original model.

These methods are usually understood and implemented in terms of affine combinations/weighted averages. The taking of an affine combination in Euclidean space is a fundamental operation and very useful for efficient geometric processing. As a result, redefining weighted averages within the manifold, spherical, and Riemennian spaces have been studied in several previous works [29, 30].

Affine combinations on the sphere have been approached via iterative optimization [8]. However, since the exact results of the weighted averages in this method are not known *a priori* (due to iterative solving of the optimization), we cannot develop a loss-less multiresolution scheme based on this method in the approximating case.

The coefficients of an affine combination may be used as barycentric coordinates to describe a point with respect to a set of polygon vertices. The spherical barycentric coordinates of a point p inside a spherical triangle may be calculated using the method described in [17], or for a point p inside a spherical polygon using the work of [31]. In [17], the resulting barycentric coordinates may be used to represent p as a linear combination of the vertices of the spherical triangle. Unlike barycentric coordinates in Euclidean space, the coefficients do not sum to unity and are dependent on properties of the spherical triangle.

Subdivision for curves on general manifolds has been proposed in [12, 32, 33, 34] and for spheres in particular in [13, 35]. However, these works do not present corresponding spherical reverse subdivision or multiresolution schemes. Similarly, the well-known Ramer-Douglas-Peucker algorithm [36] can be used to reduce the number of points in a curve, but is a simple downsampling and does not support loss-less reconstruction of the original curve.

In [11], the authors define multiresolution schemes on general manifolds using the exponential map. They focus particularly on interpolating and midpoint-interpolating subdivision schemes, for which perfect reconstruction may be achieved and guaranteed. In the paper, the authors note that it is not clear how to achieve perfect reconstruction in the approximating case. Developing multiresolution for neither interpolating nor midpoint-interpolating scheme remains an important task.

Multiresolution for spherical domains has also been proposed in wavelet form [37, 38]. These works do not represent spherical curves explicitly — they must first be approximated using a wavelet function. Consequently, the multiresolution in these works is not directly defined on spherical curves but rather on the parametrization of the sphere.

3. Spherical Lane-Riesenfeld

Instead of affine combinations, it is possible to implement subdivision using simpler geometric operations that are easier to translate into other spaces. In particular, there exists a construction for B-Spline subdivision schemes of arbitrary degree based on *repeated averaging* that uses only midpoint operations, known as the Lane-Riesenfeld algorithm [15]. The generalization of the Lane-Riesenfeld algorithm to spherical space is noted in [13], which we reiterate here, before extending the algorithm to a spherical multiresolution framework.

The construction for a B-Spline subdivision scheme of degree k in Euclidean space operates as follows. For each application of the subdivision scheme to a given curve, the curve's vertices are first duplicated, and then k averaging steps are applied to the curve. The averaging step moves each vertex to its midpoint with its consecutive neighbour. At the limit of repeated applications of this method, the vertices converge to a B-Spline curve of degree k with C^{k-1} continuity.

In mathematical notation, let P_{LR} be the desired subdivision transformation of degree k, c be a vector of coarse points, and f be the vector of fine points resulting from subdivision on c. Then, $f = P_{LR}(c)$ and $P_{LR} = S^k \circ P_H$ where P_H is a point duplication operation (and, in fact, a Haar subdivision operation [24]) with matrix form (here shown for 3 coarse vertices)

$$P_H = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

and S is the averaging transformation with matrix form

$$S = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & 0 & 0 & \frac{1}{2} \end{bmatrix}.$$

Hence, given two consecutive points p_i and p_{i+1}, $S(p_i) = \frac{1}{2}p_i + \frac{1}{2}p_{i+1}$.

The generalization to spherical space presented in [13] replaces the midpoint operation with a geodesic midpoint operation, so that S becomes $\mathcal{S}(p_i) = \text{SLERP}(p_i, p_{i+1}, \frac{1}{2})$ and $\mathcal{P}_{LR} = \mathcal{S}^k \circ P_H$.

Note that the subdivision is only valid if the angle between consecutive coarse points c_i and c_{i+1} is less than 180° for all i. If the angle is equal to 180°, then the geodesic midpoint is undefined, and if the angle is greater than 180°, then this angle cannot be computed directly from c_i and c_{i+1} and will need to be stored. However, this case is rarely of interest.

4. Reversing Lane-Riesenfeld with Atomic Operations

While this generalized Lane-Riesenfeld construction makes it possible to achieve forward subdivision on the surface of the sphere, unfortunately a similar construction for efficient reverse subdivision (an essential component of multiresolution) does not exist, as the averaging transformations S and $\mathcal{S}$ are not invertible. This is due to the fact that, given an even number of (geodesic) midpoints, the set of points with these (geodesic) midpoints is not unique.

In [16], Sadeghi and Samavati present a class of local (i.e. banded) fairing matrices with local (i.e. banded) inverses. As noted in their work, an approximation of S with local inverse can be created by employing a two-pass approach, wherein every other point is fixed during an averaging pass, and can be used to construct a modified version of the Lane-Riesenfeld algorithm. Note that the authors do not present corresponding reverse subdivision or multiresolution operations for their

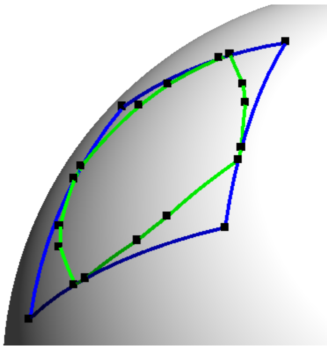


Figure 2: The spherical chasing game as applied to a spherical curve. A curve is shown before (in blue) and after (in green) two applications of degree 1 subdivision. Notice the significant skewing and clustering of vertices.

scheme, as the focus of the work is on smoothing curves in a reversible manner.

There are several benefits to having a banded/local inverse. The entries of global inverse matrices change as the input size changes, whereas the entries of local inverses are consistent. This consistency allows us to decompose the affine combinations represented by such inverses into series of two-point interpolations/SLERPs that can be applied to all points and are guaranteed to invert the original fairing operation. Furthermore, the local nature of a banded inverse can be exploited to apply the inverse transformation in linear time (with respect to input size), rather than quadratic time.

We introduce in this section a generalization of Sadeghi and Samavati's modified Lane-Riesenfeld algorithm to spherical space, which we refer to as the *spherical chasing game* to distinguish between this scheme and spherical Lane-Riesenfeld. We denote it as $\mathcal{P}_{CG}$. Making use of the same basic principles behind this scheme, we later develop a novel subdivision scheme that addresses its main problems and describe how multiresolution may be achieved.

The algorithm begins with a duplication step P_H followed by k iterations of a spherical averaging step $\mathcal{F}$, hence $f = \mathcal{P}_{CG}(c) = \mathcal{F}^k \circ P_H(c)$. The averaging step in this case uses a two-pass approach in which half the vertices at a time are fixed in place. That is, $\mathcal{F} = \mathcal{F}_1 \circ \mathcal{F}_0$, where $\mathcal{F}_0$ and $\mathcal{F}_1$ are fairing operations with local inverses $\mathcal{F}_0^{-1}$ and $\mathcal{F}_1^{-1}$ that move only half the points at a time.

The action of $\mathcal{F}_0$ is to move each vertex with an even index to its geodesic midpoint with its consecutive neighbour, and $\mathcal{F}_1$ does the same but for vertices with odd index. Hence,

$$\mathcal{F}_0(p_i) = \begin{cases} \mathrm{SLERP}(p_i, p_{i+1}, \frac{1}{2}) & \text{if } i \text{ is even,} \\ p_i & \text{otherwise,} \end{cases}$$

$$\mathcal{F}_1(p_i) = \begin{cases} \mathrm{SLERP}(p_i, p_{i+1}, \frac{1}{2}) & \text{if } i \text{ is odd,} \\ p_i & \text{otherwise.} \end{cases}$$

$\mathcal{F}_0^{-1}$ and $\mathcal{F}_1^{-1}$ undo the effects of $\mathcal{F}_0$ and $\mathcal{F}_1$ (i.e. $\mathcal{F}_0^{-1} \circ \mathcal{F}_0(p) = p$ and $\mathcal{F}_1^{-1} \circ \mathcal{F}_1(p) = p$) via

$$\mathcal{F}_0^{-1}(p_i) = \begin{cases} \mathrm{SLERP}(p_i, p_{i+1}, -1) & \text{if } i \text{ is even,} \\ p_i & \text{otherwise,} \end{cases}$$

$$\mathcal{F}_1^{-1}(p_i) = \begin{cases} \mathrm{SLERP}(p_i, p_{i+1}, -1) & \text{if } i \text{ is odd,} \\ p_i & \text{otherwise.} \end{cases}$$

These can be represented in a more compact form (where $j \in \{0, 1\}$):

$$\mathcal{F}_j(p_i) = \begin{cases} \mathrm{SLERP}(p_i, p_{i+1}, \frac{1}{2}) & \text{if } i \bmod 2 = j, \\ p_i & \text{otherwise} \end{cases}$$

and

$$\mathcal{F}_j^{-1}(p_i) = \begin{cases} \mathrm{SLERP}(p_i, p_{i+1}, -1) & \text{if } i \bmod 2 = j, \\ p_i & \text{otherwise.} \end{cases}$$

The main problem with $\mathcal{P}_{CG}$ is the unsymmetric nature of the resulting subdivision, causing curves to skew in one direction and distribute vertices unevenly (see Figure 2).

5. Spherical Subdivision

A more symmetric scheme with better vertex distribution than $\mathcal{P}_{CG}$ can be achieved by using a more symmetric local fairing operation with local inverse. Our choice of fairing operation is motivated by a simple observation. Consider the Lane-Riesenfeld algorithm in Euclidean space, with $f = P_{LR}(c) = S^k \circ P_H(c)$ and

$$S = \begin{bmatrix} \ddots & & & & & \\ & \frac{1}{2} & \frac{1}{2} & 0 & 0 & \\ & 0 & \frac{1}{2} & \frac{1}{2} & 0 & \\ & 0 & 0 & \frac{1}{2} & \frac{1}{2} & \\ & & & & & \ddots \end{bmatrix}.$$

Note that S is not invertible. By grouping pairs of S together, we find that

$$f = \begin{cases} (S \circ S)^{\frac{k}{2}} \circ P_H(c) & \text{when } k \text{ is even,} \\ (S \circ S)^{\frac{k-1}{2}} \circ S \circ P_H(c) & \text{when } k \text{ is odd} \end{cases}$$

and

$$S \circ S = \begin{bmatrix} \ddots & & & & & & \\ & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 & 0 & \\ & 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 & \\ & 0 & 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & \\ & & & & & & \ddots \end{bmatrix}.$$

Note that $S \circ P_H$ is the linear (or Faber; see [39]) subdivision operator, which we will denote as P_F ($\mathcal{P}_F$ in the spherical case), that introduces a midpoint between each pair of consecutive points (geodesic midpoint in the spherical case).

Pairing instances of S into instances of $S^2 = S \circ S$ allows us to split P_{LR} into two different schemes: one for even degree subdivisions and one for odd degree subdivisions. In even degree subdivisions, which are dual subdivision schemes [27], we can first apply Haar subdivision (P_H) followed by $\frac{k}{2}$ applications of averaging step S^2. In odd degree subdivisions, which

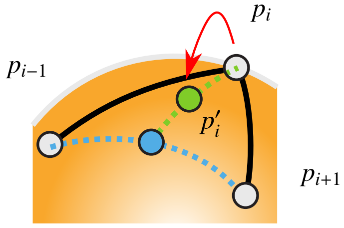


Figure 3: The modified Laplacian smoothing operator in spherical space moves a point p_i halfway to the geodesic midpoint $m_i = \mathrm{SLERP}(p_{i-1}, p_{i+1}, \frac{1}{2})$ of its neighbours.

are primal schemes [27], we can first apply Faber subdivision (P_F) followed by $\frac{k-1}{2}$ applications of averaging step S^2.

It can be seen that S^2 is a discrete Laplacian smoothing operation, whose action is to move each vertex halfway toward the midpoint of its neighbours ($\frac{1}{2}p_{i-1} + \frac{1}{2}p_{i+1}$). In the case of $\mathcal{S}^2$, these would be geodesic midpoints (see Figure 3). Note that, as a product of singular matrices, S^2 is also not invertible. As in Section 4, the averaging step $\mathcal{S}^2$ can be modified to be invertible with local inverse on the sphere by employing a two-pass approach. To accomodate the fundamental differences between primal and dual schemes, we employ different approximations of $\mathcal{S}^2$ in the odd and even degree cases.

5.1. Odd Degree Subdivision

In the case of primal subdivision schemes like P_{LR} with odd k, which map coarse vertices to subdivided vertices, it makes sense to fix vertices of the curve while performing the two passes of the averaging step. Hence, we can replace $\mathcal{S}^2$ with the modified Laplacian smoothing operator described in [16] as generalized to spherical space (see Figure 3). In odd degree subdivisions, $\mathcal{S}^2$ is replaced by $\mathcal{G} = \mathcal{G}_1 \circ \mathcal{G}_0$, where $\mathcal{G}_0$ and $\mathcal{G}_1$ are spherical fairing operations with local inverses $\mathcal{G}_0^{-1}$ and $\mathcal{G}_1^{-1}$ that operate on only half the points at a time (see Figure 4).

The action of $\mathcal{G}_0$ is to move each vertex with an even index to its geodesic midpoint with the geodesic midpoint of its neighbours, and $\mathcal{G}_1$ does the same but for vertices with odd index. That is,

$$\mathcal{G}_j(p_i) = \begin{cases} \mathrm{SLERP}(p_i, m_i, \frac{1}{2}) & \text{if } i \bmod 2 = j, \\ p_i & \text{otherwise,} \end{cases}$$

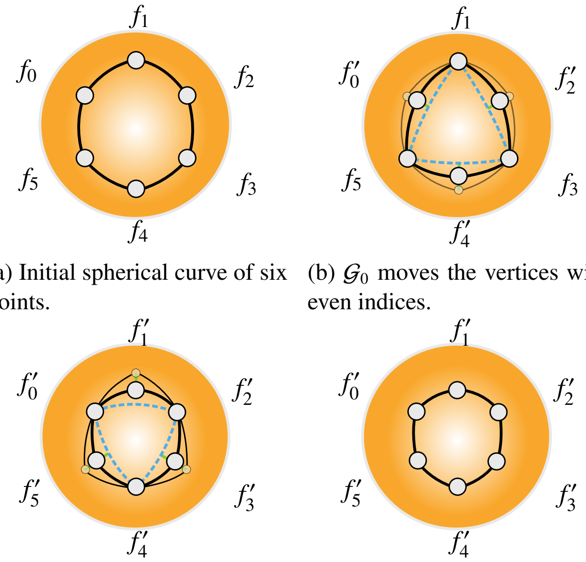
where $m_i = \mathrm{SLERP}(p_{i-1}, p_{i+1}, \frac{1}{2})$.

These can be generalized using weighting parameters, similar to the tension parameter of [40], $\{w_0, w_1, w_2, \ldots, w_{k-2}\}$ $(0 \leq w_j < 1)$ in order to vary the position of p_i along the great circle arc between p_i and m_i. Here, $j \in \{0, 1, 2, \ldots, k-2\}$ and

$$\mathcal{G}_j(p_i) = \begin{cases} \mathrm{SLERP}(p_i, m_i, w_j) & \text{if } i \bmod 2 = j \bmod 2, \\ p_i & \text{otherwise.} \end{cases}$$

Thus, when k is odd we define our spherical subdivision transformation $f = \mathcal{P}_{MR}(c)$ to be

$$\mathcal{P}_{MR} = \mathcal{G}_{k-2} \circ \cdots \circ \mathcal{G}_1 \circ \mathcal{G}_0 \circ \mathcal{P}_F.$$



(a) Initial spherical curve of six points.

(b) $\mathcal{G}_0$ moves the vertices with even indices.

(c) $\mathcal{G}_1$ moves the vertices with odd indices.

(d) Final curve after one averaging step.

Figure 4: Illustrative example of the averaging step $\mathcal{G} = \mathcal{G}_1 \circ \mathcal{G}_0$ applied to the vertices of a spherical curve.

In Euclidean space, the G_j have matrix form (here shown for 6 points)

$$G_j = \begin{bmatrix} 1-w_j & \frac{w_j}{2} & 0 & 0 & 0 & \frac{w_j}{2} \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{w_j}{2} & 1-w_j & \frac{w_j}{2} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{w_j}{2} & 1-w_j & \frac{w_j}{2} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

when j is even and

$$G_j = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{w_j}{2} & 1-w_j & \frac{w_j}{2} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{w_j}{2} & 1-w_j & \frac{w_j}{2} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ \frac{w_j}{2} & 0 & 0 & 0 & \frac{w_j}{2} & 1-w_j \end{bmatrix}$$

when j is odd.

5.2. Even Degree Subdivision

In the case of dual subdivision schemes like P_{LR} with even k, which map coarse edges to subdivided edges, it makes sense to fix the midpoints of curve edges while performing the two passes of the averaging step. Hence, we can replace $\mathcal{S}^2$ with an edge shrinking operator as generalized to spherical space (see Figure 5). In even degree subdivisions, $\mathcal{S}^2$ is replaced by $\mathcal{H} = \mathcal{H}_1 \circ \mathcal{H}_0$, where $\mathcal{H}_0$ and $\mathcal{H}_1$ are spherical fairing operations with local inverses $\mathcal{H}_0^{-1}$ and $\mathcal{H}_1^{-1}$ that operate on all points such that the midpoints of half of the edges are preserved (see Figure 6).

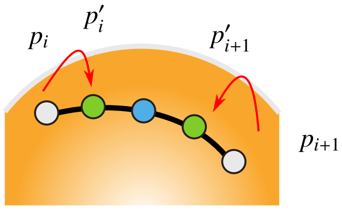
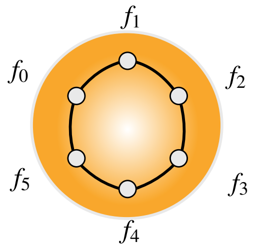
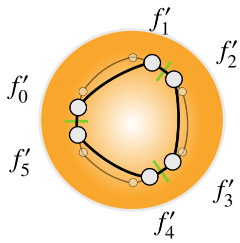


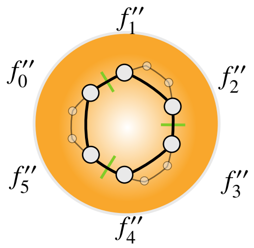
Figure 5: The edge shrinking operator in spherical space moves the edge endpoints p_i and p_{i+1} halfway to their geodesic midpoint $m_i = \text{SLERP}(p_i, p_{i+1}, \frac{1}{2})$.



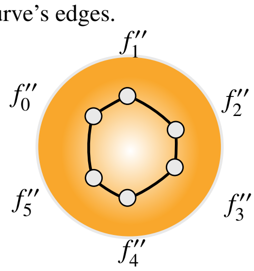
(a) Initial spherical curve of six points.



(b) $\mathcal{H}_0$ shrinks half of the curve's edges.



(c) $\mathcal{H}_1$ shrinks the other edges.



(d) Final curve after one averaging step.

Figure 6: Illustrative example of the averaging step $\mathcal{H} = \mathcal{H}_1 \circ \mathcal{H}_0$ applied to the vertices of a spherical curve.

The action of $\mathcal{H}_0$ is to shrink half of the curve's edges by moving each's endpoints half of the way towards their geodesic midpoint, and $\mathcal{H}_1$ does the same but for the other edges. That is,

$$\mathcal{H}_j(p_i) = \begin{cases} \text{SLERP}(p_i, p_{i+2j-1}, \frac{1}{4}) & \text{if } i \bmod 2 = j, \\ \text{SLERP}(p_i, p_{i-2j+1}, \frac{1}{4}) & \text{otherwise,} \end{cases}$$

where $m_i = \text{SLERP}(p_{i-1}, p_{i+1}, \frac{1}{2})$.

These too can be generalized using weighting parameters $\{w_0, w_1, w_2, \ldots, w_{k-1}\}$ $(0 \le w_j < 1)$ in order to vary the position of p_i and $p_{i\pm1}$ along the great circle arc between p_i and $p_{i\pm1}$. Here, $j \in \{0, 1, 2, \ldots, k-1\}$ and

$$\mathcal{H}_j(p_i) = \begin{cases} \text{SLERP}(p_i, p_{i+x}, \frac{w_j}{2}) & \text{if } i \bmod 2 = j \bmod 2, \\ \text{SLERP}(p_i, p_{i-x}, \frac{w_j}{2}) & \text{otherwise.} \end{cases}$$

where $x = -1$ if j is even and $x = +1$ if j is odd. Thus, our spherical subdivision transformation $f = \mathcal{P}_{MR}(c)$ is piecewise defined to be

$$\mathcal{P}_{MR} = \begin{cases} \mathcal{H}_{k-1} \circ \cdots \circ \mathcal{H}_1 \circ \mathcal{H}_0 \circ P_H & \text{when } k \text{ is even,} \\ \mathcal{G}_{k-2} \circ \cdots \circ \mathcal{G}_1 \circ \mathcal{G}_0 \circ \mathcal{P}_F & \text{when } k \text{ is odd} \end{cases}$$

In Euclidean space, the H_j have matrix form (here shown for 6 points)

$$H_j = \begin{bmatrix} 1-\frac{w_j}{2} & 0 & 0 & 0 & 0 & \frac{w_j}{2} \\ 0 & 1-\frac{w_j}{2} & \frac{w_j}{2} & 0 & 0 & 0 \\ 0 & \frac{w_j}{2} & 1-\frac{w_j}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-\frac{w_j}{2} & \frac{w_j}{2} & 0 \\ 0 & 0 & 0 & \frac{w_j}{2} & 1-\frac{w_j}{2} & 0 \\ \frac{w_j}{2} & 0 & 0 & 0 & 0 & 1-\frac{w_j}{2} \end{bmatrix}$$

when j is even and

$$H_j = \begin{bmatrix} 1-\frac{w_j}{2} & \frac{w_j}{2} & 0 & 0 & 0 & 0 \\ \frac{w_j}{2} & 1-\frac{w_j}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1-\frac{w_j}{2} & \frac{w_j}{2} & 0 & 0 \\ 0 & 0 & \frac{w_j}{2} & 1-\frac{w_j}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1-\frac{w_j}{2} & \frac{w_j}{2} \\ 0 & 0 & 0 & 0 & \frac{w_j}{2} & 1-\frac{w_j}{2} \end{bmatrix}$$

when j is odd.

6. Spherical Reverse Subdivision

A property of the $\mathcal{G}_j$ and $\mathcal{H}_j$ as defined in Section 5 is that they have local inverses $\mathcal{G}_j^{-1}$ and $\mathcal{H}_j^{-1}$ for all j. Hence, it becomes possible to undo the averaging steps. If $w_j = \frac{1}{2}$, these functions are given by

$$\mathcal{G}_j^{-1}(p_i) = \begin{cases} \text{SLERP}(p_i, m_i, -1) & \text{if } i \bmod 2 = j \bmod 2, \\ p_i & \text{otherwise,} \end{cases}$$

where, again, $m_i = \text{SLERP}(p_{i-1}, p_{i+1}, \frac{1}{2})$, and

$$\mathcal{H}_j^{-1}(p_i) = \begin{cases} \text{SLERP}(p_i, p_{i+x}, -\frac{1}{2}) & \text{if } i \bmod 2 = j \bmod 2, \\ \text{SLERP}(p_i, p_{i-x}, -\frac{1}{2}) & \text{otherwise,} \end{cases}$$

where $x = -1$ if j is even and $x = +1$ if j is odd. See Figures 7 and 8 for illustrations.

In general, given weighting parameters $0 \le w_j < 1$,

$$\mathcal{G}_j^{-1}(p_i) = \begin{cases} \text{SLERP}(p_i, m_i, \frac{w_j}{w_j - 1}) & \text{if } i \bmod 2 = j \bmod 2, \\ p_i & \text{otherwise.} \end{cases}$$

and

$$\mathcal{H}_j^{-1}(p_i) = \begin{cases} \text{SLERP}(p_i, p_{i+x}, \frac{w_j}{2w_j - 2}) & \text{if } i \bmod 2 = j \bmod 2, \\ \text{SLERP}(p_i, p_{i-x}, \frac{w_j}{2w_j - 2}) & \text{otherwise,} \end{cases}$$

After undoing the averaging steps, we must also undo the Haar subdivision P_H or the Faber subdivision $\mathcal{P}_F$. While these do not have inverses, they have corresponding multiresolution schemes (wavelet transforms) in Euclidean space that may be easily generalized to the sphere. The Euclidean reverse Haar scheme replaces each pair of vertices with their midpoint ($c_i =$

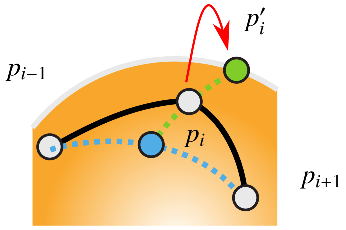


Figure 7: Inverse of the modified Laplacian smoothing operator in spherical space.

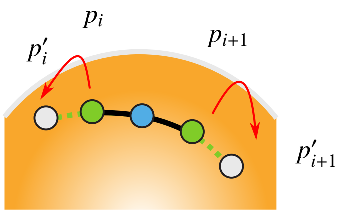


Figure 8: Inverse of the edge shrinking operator in spherical space.

$\frac{1}{2}f_{2i} + \frac{1}{2}f_{2i+1})$ [24], which in spherical space corresponds to replacing each pair of vertices with their geodesic midpoint. Although alternatives exist, reversing the Faber scheme may be accomplished by discarding every other point ($c_i = f_{2i}$). We will denote the reverse P_H operation as $\mathcal{A}_H$ and the reverse $\mathcal{P}_F$ operation as $\mathcal{A}_F$. Notice that $\mathcal{A}_H$ and $\mathcal{A}_F$ are downsampling operations.

We now define a reverse subdivision transformation $c = \mathcal{A}_{MR}(f)$ for the subdivision operation $\mathcal{P}_{MR}$ described in Section 5, where

$$\mathcal{A}_{MR} = \begin{cases} \mathcal{A}_H \circ \mathcal{H}_0^{-1} \circ \mathcal{H}_1^{-1} \circ \cdots \circ \mathcal{H}_{k-1}^{-1} & \text{when } k \text{ is even,} \\ \mathcal{A}_F \circ \mathcal{G}_0^{-1} \circ \mathcal{G}_1^{-1} \circ \cdots \circ \mathcal{G}_{k-2}^{-1} & \text{when } k \text{ is odd} \end{cases}$$

Note that we restrict f such that the angle between consecutive points f_i and f_{i+1} must be less than 90° for all i, as otherwise the reverse scheme may introduce c_i and c_{i+1} for some i with an angle $\theta \geq 180°$ between them, which invalidates the forward subdivision operation.

7. Spherical Multiresolution

Since $\mathcal{G}_j$ and $\mathcal{H}_j$ are invertible for all j, high resolution details of the curve are only lost during the application of the reverse operations $\mathcal{A}_H$ and $\mathcal{A}_F$. Hence, it is possible to achieve multiresolution in this case using the foundations of Haar wavelets and Faber wavelets. See Algorithms 1 to 4 for our framework's pseudocode.

Let $f = [f_0 \dots f_{m-1}]^T$ be the fine points before reverse subdivision and $c = \mathcal{A}_{MR}(f) = [c_0 \dots c_{n-1}]^T$ be the coarse points resulting from reverse subdivision on f. In Euclidean space, the Haar detail vectors are found as $d_i = \frac{1}{2}f_{2i+1} - \frac{1}{2}f_{2i}$ [24]. Then, given that $c_i = \frac{1}{2}f_{2i} + \frac{1}{2}f_{2i+1}$, it should be clear that $f_{2i} = c_i - d_i$ and $f_{2i+1} = c_i + d_i$. In spherical space, we can generalize this so that we find detail *rotations*, rather than detail vectors, given by

$$\begin{aligned} d_i &= \text{half the rotation from } f_{2i} \rightarrow f_{2i+1} \\ &= \text{a rotation of angle } \frac{\cos^{-1}(f_{2i} \cdot f_{2i+1})}{2} \text{ about axis } f_{2i} \times f_{2i+1}, \end{aligned}$$

while the reconstruction is given by

$$\begin{aligned} f_{2i} &= c_i \text{ rotated by } d_i^{-1}, \\ f_{2i+1} &= c_i \text{ rotated by } d_i. \end{aligned}$$

The transformation that finds the detail rotations, which we denote as $\mathcal{B}_H$, takes place after the inverse averaging steps have been applied but before the reverse Haar operation $\mathcal{A}_H$. The transformation that restores the details (i.e. rotates the points), which we call $\mathcal{Q}_H$, takes place after duplication but before the averaging steps.

The Faber case is similar. For the given reverse subdivision operation, the details can be found as $d_i = f_{2i+1} - (\frac{1}{2}f_{2i} + \frac{1}{2}f_{2i+2})$. Given that $c_i = f_{2i}$, it can be seen that $f_{2i} = c_i$ and $f_{2i+1} = \frac{1}{2}c_i + \frac{1}{2}c_{i+1} + d_i$. In spherical space, the details are

$$\begin{aligned} d_i &= \text{the rotation from } m_i \rightarrow f_{2i+1} \\ &= \text{a rotation of angle } \cos^{-1}(m_i \cdot f_{2i+1}) \text{ about axis } m_i \times f_{2i+1} \end{aligned}$$

where $m_i = \text{SLERP}(f_{2i}, f_{2i+2}, \frac{1}{2})$, and the reconstruction is defined by

$$\begin{aligned} f_{2i} &= c_i, \\ f_{2i+1} &= \text{SLERP}(c_i, c_{i+1}, \tfrac{1}{2}) \text{ rotated by } d_i. \end{aligned}$$

As before, we denote by $\mathcal{B}_F$ and $\mathcal{Q}_F$ the transformations that calculate and restore the detail rotations, respectively.

Now we may define our multiresolution framework on spherical curves. The reconstruction step determines the fine points

$$f = \begin{cases} \mathcal{H}_{k-1} \circ \cdots \circ \mathcal{H}_1 \circ \mathcal{H}_0 \circ \mathcal{Q}_H(P_H(c), d) & \text{when } k \text{ is even,} \\ \mathcal{G}_{k-2} \circ \cdots \circ \mathcal{G}_1 \circ \mathcal{G}_0 \circ \mathcal{Q}_F(\mathcal{P}_F(c), d) & \text{when } k \text{ is odd} \end{cases}$$

and the decomposition step determines the coarse points

$$c = \begin{cases} \mathcal{A}_H \circ \mathcal{H}_0^{-1} \circ \mathcal{H}_1^{-1} \circ \cdots \circ \mathcal{H}_{k-1}^{-1}(f) & \text{when } k \text{ is even,} \\ \mathcal{A}_F \circ \mathcal{G}_0^{-1} \circ \mathcal{G}_1^{-1} \circ \cdots \circ \mathcal{G}_{k-2}^{-1}(f) & \text{when } k \text{ is odd} \end{cases}$$

and the detail rotations

$$d = \begin{cases} \mathcal{B}_H \circ \mathcal{H}_0^{-1} \circ \mathcal{H}_1^{-1} \circ \cdots \circ \mathcal{H}_{k-1}^{-1}(f) & \text{when } k \text{ is even,} \\ \mathcal{B}_F \circ \mathcal{G}_0^{-1} \circ \mathcal{G}_1^{-1} \circ \cdots \circ \mathcal{G}_{k-2}^{-1}(f) & \text{when } k \text{ is odd.} \end{cases}$$

These operations, which operate in the spherical domain, are both simple and efficient.

Note that the detail rotations can each be represented compactly as a vector of three components whose direction indicates the axis of rotation and whose magnitude is equal to the angle of rotation. Hence, we have achieved a simple and efficient multiresolution framework on the sphere without increasing the memory footprint.

Algorithm 1 Pseudocode for even degree spherical decomposition step.

DECOMPOSE-EVEN:
Input:
- fine points $F[0 \ldots m-1]$,
- weights $W[0 \ldots k-1]$

Output:
- coarse points $C[0 \ldots \frac{m}{2}-1]$,
- detail rotations $D[0 \ldots \frac{m}{2}-1]$

```
P := F
for j := k − 1 to 0 step −2 do
  for i := 0 to m − 2 step 2 do
    p := P[i]
    P[i] := SLERP(p, P[i + 1], W[j]/(2W[j]−2))
    P[i + 1] := SLERP(P[i + 1], p, W[j]/(2W[j]−2))
  end for
  for i := 1 to m − 1 step 2 do
    p := P[i]
    P[i] := SLERP(p, P[i + 1], W[j−1]/(2W[j−1]−2))
    P[i + 1] := SLERP(P[i + 1], p, W[j−1]/(2W[j−1]−2))
  end for
end for
for i := 0 to m − 2 step 2 do
  C[i/2] := SLERP(P[i], P[i + 1], 1/2)
  D[i/2] := half the rotation from P[i] → P[i + 1]
end for
return C, D
```

Algorithm 2 Pseudocode for even degree spherical reconstruction step.

RECONSTRUCT-EVEN:
Input:
- coarse points $C[0 \ldots n-1]$,
- detail rotations $D[0 \ldots n-1]$,
- weights $W[0 \ldots k-1]$

Output:
- fine points $F[0 \ldots 2n-1]$

```
for i := 0 to n − 1 do
  F[2i] := C[i] rotated by D[i]^−1
  F[2i + 1] = C[i] rotated by D[i]
end for
for j := 0 to k − 1 step 2 do
  for i := 1 to 2n − 1 step 2 do
    p := P[i]
    P[i] := SLERP(p, P[i + 1], W[j]/2)
    P[i + 1] := SLERP(P[i + 1], p, W[j]/2)
  end for
  for i := 0 to 2n − 2 step 2 do
    p := P[i]
    P[i] := SLERP(p, P[i + 1], W[j+1]/2)
    P[i + 1] := SLERP(P[i + 1], p, W[j+1]/2)
  end for
end for
return F
```

Algorithm 3 Pseudocode for odd degree spherical decomposition step.

DECOMPOSE-ODD:
Input:
- fine points $F[0 \ldots m-1]$,
- weights $W[0 \ldots k-2]$

Output:
- coarse points $C[0 \ldots \frac{m}{2}-1]$,
- detail rotations $D[0 \ldots \frac{m}{2}-1]$

```
P := F
for j := k − 2 to 0 step −2 do
  for i := 1 to m − 1 step 2 do
    mid = SLERP(P[i − 1], P[i + 1], 1/2)
    P[i] := SLERP(P[i], mid, W[j]/(W[j]−1))
  end for
  for i := 0 to m − 2 step 2 do
    mid = SLERP(P[i − 1], P[i + 1], 1/2)
    P[i] := SLERP(P[i], mid, W[j−1]/(W[j−1]−1))
  end for
end for
for i := 0 to m − 2 step 2 do
  mid = SLERP(P[i], P[i + 2], 1/2)
  C[i/2] := P[i]
  D[i/2] := the rotation from mid → P[i + 1]
end for
return C, D
```

Algorithm 4 Pseudocode for odd degree spherical reconstruction step.

RECONSTRUCT-ODD:
Input:
- coarse points $C[0 \ldots n-1]$,
- detail rotations $D[0 \ldots n-1]$,
- weights $W[0 \ldots k-2]$

Output:
- fine points $F[0 \ldots 2n-1]$

```
for i := 0 to n − 1 do
  F[2i] := C[i]
  F[2i + 1] := SLERP(C[i], C[i + 1], 1/2) rotated by D[i]
end for
for j := 0 to k − 2 step 2 do
  for i := 0 to 2n − 2 step 2 do
    mid = SLERP(F[i − 1], F[i + 1], 1/2
    F[i] := SLERP(F[i], mid, W[j])
  end for
  for i := 1 to 2n − 1 step 2 do
    mid = SLERP(F[i − 1], F[i + 1], 1/2
    F[i] := SLERP(F[i], mid), W[j + 1])
  end for
end for
return F
```

8. Analysis

Two important questions when dealing with any subdivision scheme are that of whether or not the subdivided vertices will converge to a curve at the limit of repeated applications of the subdivision, and what level of continuity the limit curve would have. While a thorough discussion of the subdivision analysis of our scheme goes beyond the scope of this paper, we briefly touch on these issues in the following section. See [**?**] for a more thorough analysis.

Another important question, when dealing with a mulitresolution scheme, is that of whether or not the wavelets are biorthogonal. We prove, in the second half of the section, that this is indeed the case for our multiresolution scheme.

8.1. Subdivision Analysis

The work of [27] discusses techniques to analyse the convergence and continuity of subdivision schemes in Euclidean space using the scheme's subdivision mask and generating function. A consequence of including weighting parameters w_j is that the subdivision matrix of our scheme

$$P_{MR} = \begin{cases} H_{k-1} \circ \cdots \circ H_1 \circ H_0 \circ P_H & \text{when } k \text{ is even,} \\ G_{k-2} \circ \cdots \circ G_1 \circ G_0 \circ P_F & \text{when } k \text{ is odd} \end{cases}$$

(and by extension the mask and generating function) changes as the weights w_j change.

For example, taking weight values $w_0 = \frac{2}{3}$ and $w_1 = \frac{1}{4}$ with $k = 2$ produces the subdivision mask $[\frac{1}{24}, \frac{7}{24}, \frac{2}{3}, \frac{2}{3}, \frac{7}{24}, \frac{1}{24}]$ with generating function $S(z) = \frac{1}{24} + \frac{7}{24}z + \frac{2}{3}z^2 + \frac{2}{3}z^3 + \frac{7}{24}z^4 + \frac{1}{24}z^5$. Using division by $(1 + z)$, we can find 3 difference schemes for $S(z)$ with row sums less than 1, which indicates G^2 continuity of the limit curve [27]. These same weights with $k = 3$ produce the mask $[\frac{1}{48}, \frac{1}{6}, \frac{23}{48}, \frac{2}{3}, \frac{23}{48}, \frac{1}{6}, \frac{1}{48}]$, for which G^3 continuity can be shown.

Interestingly, weight values of $w_0 = \frac{1}{2}$ and $w_1 = 0$ with $k = 2$ produce the subdivision mask $[\frac{1}{4}, \frac{3}{4}, \frac{3}{4}, \frac{1}{4}]$ of Chaikin subdivision [2], which is known to converge to a 2nd degree B-Spline curve at the limit with G^1 continuity. These same weight values with $k = 3$ produce the subdivision mask $[\frac{1}{8}, \frac{1}{2}, \frac{3}{4}, \frac{1}{2}, \frac{1}{8}]$ of cubic subdivision (see [4]), which converges to a 3rd degree B-Spline curve at the limit with G^2 continuity.

Higher degree B-Spline subdivision masks can be obtained using different weights, illustrating that our construction is more powerful than the Lane-Riesenfeld algorithm not only in terms of providing built-in multiresolution capabilities, but also in supporting a class of subdivision schemes including at least some (and potentially all) B-Spline subdivision schemes.

We expect that the limit curves resulting from our spherical subdivision scheme will have the same continuities as their Euclidean counterparts. Intuitively, since the surface of the sphere is locally isometric to a plane and is infinitely differentiable, for sufficiently close vertices of the spherical curve our spherical scheme will behave like the Euclidean scheme, and will hence have the same continuity. It remains to be shown, however, that the spherical scheme converges on the sphere (using, e.g., the works of [12, 28]), so that this condition of sufficient closeness is eventually satisfied.

8.2. Proof of Biorthogonality

Given a subdivision matrix P and detail restoration matrix Q, as well as a reverse subdivision matrix A and detail calculation matrix B, the multiresolution scheme defined by P, Q, A, and B is biorthogonal if

$$\begin{bmatrix} A \\ B \end{bmatrix} \begin{bmatrix} P & Q \end{bmatrix} = \begin{bmatrix} P & Q \end{bmatrix} \begin{bmatrix} A \\ B \end{bmatrix} = I.$$

It can be verified that both the Haar and Faber schemes described here are biorthogonal.

For our multiresolution scheme, in addition to P_{MR} (see above), we have matrices

$$Q_{MR} = \begin{cases} H_{k-1} \circ \cdots \circ H_1 \circ H_0 \circ Q_H & \text{when } k \text{ is even,} \\ G_{k-2} \circ \cdots \circ G_1 \circ G_0 \circ Q_F & \text{when } k \text{ is odd,} \end{cases}$$

$$A_{MR} = \begin{cases} A_H \circ H_0^{-1} \circ H_1^{-1} \circ \cdots \circ H_{k-1}^{-1} & \text{when } k \text{ is even,} \\ A_F \circ G_0^{-1} \circ G_1^{-1} \circ \cdots \circ G_{k-2}^{-1} & \text{when } k \text{ is odd,} \end{cases}$$

and

$$B_{MR} = \begin{cases} B_H \circ H_0^{-1} \circ H_1^{-1} \circ \cdots \circ H_{k-1}^{-1} & \text{when } k \text{ is even,} \\ B_F \circ G_0^{-1} \circ G_1^{-1} \circ \cdots \circ G_{k-2}^{-1} & \text{when } k \text{ is odd.} \end{cases}$$

Now, when k is even,

$$\begin{aligned} \begin{bmatrix} A_{MR} \\ B_{MR} \end{bmatrix} \begin{bmatrix} P_{MR} & Q_{MR} \end{bmatrix} &= \begin{bmatrix} A_{MR}P_{MR} & A_{MR}Q_{MR} \\ B_{MR}P_{MR} & B_{MR}Q_{MR} \end{bmatrix} \\ &= \begin{bmatrix} A_H P_H & A_H Q_H \\ B_H P_H & B_H Q_H \end{bmatrix} \\ &= \begin{bmatrix} A_H \\ B_H \end{bmatrix} \begin{bmatrix} P_H & Q_H \end{bmatrix} \\ &= I \end{aligned}$$

and, by uniqueness of inverse,

$$\begin{bmatrix} P_{MR} & Q_{MR} \end{bmatrix} \begin{bmatrix} A_{MR} \\ B_{MR} \end{bmatrix} = I.$$

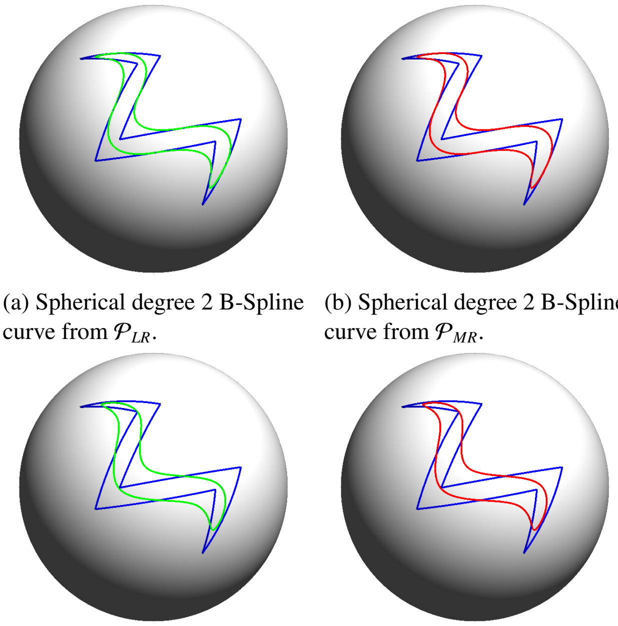
When k is odd,

$$\begin{aligned} \begin{bmatrix} A_{MR} \\ B_{MR} \end{bmatrix} \begin{bmatrix} P_{MR} & Q_{MR} \end{bmatrix} &= \begin{bmatrix} A_{MR}P_{MR} & A_{MR}Q_{MR} \\ B_{MR}P_{MR} & B_{MR}Q_{MR} \end{bmatrix} \\ &= \begin{bmatrix} A_F P_F & A_F Q_F \\ B_F P_F & B_F Q_F \end{bmatrix} \\ &= \begin{bmatrix} A_F \\ B_F \end{bmatrix} \begin{bmatrix} P_F & Q_F \end{bmatrix} \\ &= I \end{aligned}$$

and, by uniqueness of inverse,

$$\begin{bmatrix} P_{MR} & Q_{MR} \end{bmatrix} \begin{bmatrix} A_{MR} \\ B_{MR} \end{bmatrix} = I.$$

Hence, our multiresolution scheme is biorthogonal.



(a) Spherical degree 2 B-Spline curve from $\mathcal{P}_{LR}$.

(b) Spherical degree 2 B-Spline curve from $\mathcal{P}_{MR}$.

(c) Spherical degree 3 B-Spline curve from $\mathcal{P}_{LR}$.

(d) Spherical degree 3 B-Spline curve from $\mathcal{P}_{MR}$.

Figure 9: Spherical B-Spline curves resulting from spherical Lane-Riensenfeld subdivision $\mathcal{P}_{LR}$ and our subdivision scheme $\mathcal{P}_{MR}$, shown side-by-side. The original coarse curve is shown in blue.

9. Results and Comparisons

We have experimented with applying our multiresolution scheme on real geospatial vector data representing a political boundary and on manually generated spherical curves. See Figures 11 through 15 for some result images.

With weights $w_0 = \frac{1}{2}, w_1 = 0$ allowing us to recreate the subdivision filters for 2nd and 3rd degree B-Spline curves, the output of our scheme P_{MR} in Euclidean space matches the output of the Lane-Riesenfeld algorithm P_{LR} for $k = 2$ and $k = 3$. However, some amount of deviation can be expected in spherical space due to the different sequences of SLERPs used in each scheme. This deviation can be measured as the average geodesic distance between corresponding points on the subdivided curves, and in our experiments this deviation has been negligible. See Figure 9 for a side-by-side comparison of a curve subdivided with $\mathcal{P}_{MR}$ and $\mathcal{P}_{LR}$.

Figures 13 and 14 illustrate the effects of our reverse subdivision on subdivided fine curves. Using our method, the original coarse curve used to generate the subdivided fine curve can be found. In general, this is difficult to achieve, yet our method can accomplish this using only atomic operations.

As geospatial vector data can be very large in size, the speed of a multiresolution scheme applied to the data could potentially impact a geospatial application's runtime. In Table 1, we compare the runtime of our $k = 2$ scheme with the spherical Dyn-Levin scheme $\mathcal{P}_{DL}$ described in [11] (implemented using the iterative algorithm A1 from [8] with an error threshold of 10^{-7}). The values shown were calculated on a 64-bit Windows 7 machine with an Intel Core i7-4790 CPU, averaged over 100,000 runs of each scheme on a curve with vertices at $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$.

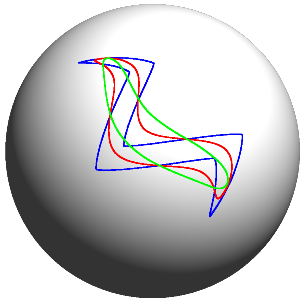


Figure 10: The coarse curve shown in Figure 9 after $k = 3$ spherical subdivision with different weight values. Weight values $w_1 = w_2 = \frac{1}{10}$ were used to generate the blue curve, $w_1 = w_2 = \frac{1}{2}$ for the red curve, and $w_1 = w_2 = \frac{9}{10}$ for the green curve.

	Runtime of $\mathcal{P}_{MR}$ (ms)	Runtime of $\mathcal{P}_{DL}$ (ms)	Average # of iterations for $\mathcal{P}_{DL}$
1st Application	0.013	0.143	4.000
2nd Application	0.024	0.109	1.000
3rd Application	0.048	0.219	1.000
4th Application	0.092	0.203	0.000
5th Application	0.183	0.409	0.000

Table 1: Runtime results from applying our $k = 2$ subdivision scheme and the spherical Dyn-Levin subdivision scheme from [11] to a curve with vertices $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$.

Note that the number of iterations for algorithm A1 converges to zero as the points become closer together and the sphere becomes locally isometric to a plane. Our scheme appears to be quite fast, taking up only half the time of the other subdivision scheme. Even in cases where algorithm A1 does not need to iterate, the cost of evaluating the iteration termination condition pushes the runtime above our scheme's.

The behaviour of the multiresolution scheme can be tweaked by varying the weight parameters w_j. Lower values for the w_j (approaching 0) will result in near-interpolation of the coarse vertices, while the reverse scheme will nearly interpolate the fine vertices and have minimal shape exaggeration (see, for example, Figure 15). Higher values for the w_j (approaching 1) will result in more straightening of the curve, while the reverse subdivision will cause more shape exaggeration. Example results are shown in Figure 10.

10. Conclusions and Future Work

We have presented a new multiresolution framework for spherical curves. Such a framework makes it possible to increase and decrease the resolution of spherical curves without leaving the spherical domain, hence avoiding distortions due to intermediate mappings. Furthermore, the framework achieves perfect reconstruction despite the subdivision being neither interpolating nor midpoint interpolating.

The key behind our construction is the use of simple geometric transformations that are generalizable to the spherical domain and make for efficient decomposition and reconstruction operations. The construction is based on a modified Lane-Riensenfeld algorithm that uses locally invertible averaging steps in place of the algorithm's non-invertible midpoint operator, and calculates details as detail rotations.

As a potential direction for future work, it would be interesting to create adaptive subdivision and reverse subdivision schemes in spherical space and/or extend our framework to other manifolds, particularly ellipsoids and geoids. Determining those manifolds with simple midpoint operations could be an important first step. Another interesting avenue of research could be to extend the Euclidean curve scheme to surfaces.

Acknowledgements

Funding for this research was provided by an NSERC CRD with our industrial collaborator, the PYXIS innovation.

References

[1] Samavati FF, Bartels RH, Olsen L. Local B-spline multiresolution with examples in iris synthesis and volumetric rendering. In: Image Pattern Recognition: Synthesis and Analysis in Biometrics; vol. 67 of *Series in Machine Perception and Artificial Intelligence*. World Scientific Publishing; 2007, p. 65–102.

[2] Chaikin GM. An algorithm for high-speed curve generation. Computer Graphics and Image Processing 1974;3(4):346–9.

[3] Catmull E, Clark J. Recursively generated B-spline surfaces on arbitrary topological meshes. Computer-Aided Design 1978;10(6):350–5.

[4] Samavati FF, Bartels RH. Multiresolution curve and surface representation: Reversing subdivision rules by least-squares data fitting. Computer Graphics Forum 1999;18(2):97–120.

[5] Bartels RH, Samavati FF. Reversing subdivision rules: Local linear conditions and observations on inner products. Journal of Computational and Applied Mathematics 2000;119(1-2):29–67.

[6] Mahdavi-Amiri A, Alderson T, Samavati F. A survey of Digital Earth. Computers & Graphics 2015;53, Part B:95–117.

[7] Freeden W, Schreiner M, Franke R. A survey on spherical spline approximation. Surveys on Mathematics for Industry 1997;7(1):29–86.

[8] Buss SR, Fillmore JP. Spherical averages and applications to spherical splines and interpolation. ACM Transactions on Graphics 2001;20(2):95–126.

[9] Snyder JP. An equal-area map projection for polyhedral globes. Cartographica 1992;29(1):10–21.

[10] do Carmo MP. Differential Geometry of Curves and Surfaces; chap. 4-6. Prentice-Hall, Inc.; 1976,.

[11] Grohs P, Wallner J. Definability and stability of multiscale decompositions for manifold-valued data. Journal of the Franklin Institute 2012;349(5):1648–64.

[12] Wallner J, Dyn N. Convergence and C1 analysis of subdivision schemes on manifolds by proximity. Computer Aided Geometric Design 2005;22(7):593–622.

[13] Schaefer S, Goldman R. Freeform curves on spheres of arbitrary dimension. In: Proceedings of Pacific Graphics 2005. 2005, p. 160–2.

[14] Shoemake K. Animating rotation with quaternion curves. In: Proceedings of SIGGRAPH 1985. 1985, p. 245–54.

[15] Lane JM, Riesenfeld RF. A theoretical development for the computer generation and display of piecewise polynomial surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence 1980;2(1):35–46.

[16] Sadeghi J, Samavati FF. Local fairing with local inverse. In: Proceedings of the 2013 Graphics Interface Conference. GI '13; 2013,.

[17] Alfeld P, Neamtu M, Schumaker LL. Bernstein-Bézier polynomials on spheres and sphere-like surfaces. Computer Aided Geometric Design 1996;13(4):333–49.

[18] Praun E, Hoppe H. Spherical parametrization and remeshing. ACM Transactions on Graphics 2003;22(3):340–9.

[19] Ghosh A, Chen T, Peers P, Wilson CA, Debevec P. Circularly polarized spherical illumination reflectometry. ACM Transactions on Graphics 2010;29(6):162:1–162:11.

[20] Goodchild MF. Discrete global grids for digital Earth. In: Proceedings of the 1st International Conference on Discrete Global Grids. 2000,.

[21] Wartell Z, Kang E, Wasilewski T, Ribarsky W, Faust N. Rendering vector data over global, multi-resolution 3D terrain. In: Proceedings of the Symposium on Data Visualisation. VISSYM '03; 2003, p. 213–22.

[22] Mahdavi-Amiri A, Bhojani F, Samavati F. One-to-two Digital Earth. In: Advances in Visual Computing; vol. 8034 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg; 2013, p. 681–92.

[23] Mallat SG. A theory for multiresolution signal decomposition: The wavelet representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 1989;11(7):674–93.

[24] Stollnitz EJ, Derose TD, Salesin DH. Wavelets for Computer Graphics: Theory and Applications. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1996. ISBN 1-55860-375-1.

[25] Garland M. Multiresolution modeling: Survey & future opportunities. In: State of the Art Reports. 1999, p. 111–31.

[26] Cashman TJ. Beyond Catmull-Clark? A survey of advances in subdivision surface methods. Computer Graphics Forum 2012;31(1):42–61.

[27] Sabin M. Analysis and Design of Univariate Subdivision Schemes; vol. 6 of *Geometry and Computing*. Springer-Verlag Berlin Heidelberg; 2010.

[28] Wallner J, Pottmann H. Smoothness analysis of subdivision schemes by proximity. Constructive Approximation 2006;24(3):289–318.

[29] Rustamov RM. Barycentric coordinates on surfaces. Computer Graphics Forum 2010;29(5):1507–16.

[30] Panozzo D, Baran I, Diamanti O, Sorkine-Hornung O. Weighted averages on surfaces. ACM Transactions on Graphics 2013;32(4):60:1–60:12.

[31] Langer T, Belyaev A, Seidel HP. Spherical barycentric coordinates. In: Proceedings of the Symposium on Geometry Processing. SGP '06; 2006, p. 81–8.

[32] Wallner J, Pottmann H. Intrinsic subdivision with smooth limits for graphics and animation. ACM Transactions on Graphics 2006;25(2):356–74.

[33] Martínez-Morera D, Velho L, Carvalho PC. Subdivision curves on surfaces and applications. In: Progress in Pattern Recognition, Image Analysis and Applications. Springer Berlin Heidelberg; 2008, p. 405–12.

[34] Estrada-Sarlabous J, Hernández-Mederos V, Martínez-Morera D, Velho L. Subdivision Curves on Surfaces with Arc-length Control. Inst. Nacional de Matemática Pura e Aplicada; 2009.

[35] Odehnal B. Subdivision algorithms for ruled surfaces. Journal for Geometry and Graphics 2008;12(1):1–18.

[36] Douglas DH, Peucker TK. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. The Canadian Cartographer 1973;10(2):112–22.

[37] Schröder P, Sweldens W. Spherical wavelets: Efficiently representing functions on the sphere. In: Proceedings of SIGGRAPH 1995. 1995, p. 161–72.

[38] Lessig C, Fiume E. SOHO: Orthogonal and symmetric Haar wavelets on the sphere. ACM Transactions on Graphics 2008;27(1):4:1–4:11.

[39] Faber G. Uber stetige funktionen. Mathematische Annalen 1909;66(1):81–94.

[40] Dyn N, Levin D, Gregory JA. A 4-point interpolatory subdivision scheme for curve design. Computer Aided Geometric Design 1987;4(4):257–68.

Appendix: Proof of Inversion

In this appendix, we prove that the operations $\mathcal{F}_j^{-1}$, $\mathcal{G}_j^{-1}$, and $\mathcal{H}_j^{-1}$ are, in fact, inverses of the operations $\mathcal{F}_j$, $\mathcal{G}_j$, and $\mathcal{H}_j$.

Lemma 1. *Given a real value $0 \leq w < 1$ and any two points p and q,*

$$SLERP\left(SLERP(p, q, w), q, \frac{w}{w-1}\right) = p.$$

Proof. Let θ be the angle between points $p' = \text{SLERP}(p, q, w)$ and q and let ψ be the angle between points p and q. Notice that, by construction of SLERP, $\theta = (1-w)\psi$.

Now,

$$\begin{aligned}
&\text{SLERP}\left(p', q, \frac{w}{w-1}\right)\\
&= \frac{\sin\left[(1-\frac{w}{w-1})\theta\right]p' + \sin\left(\frac{w}{w-1}\theta\right)q}{\sin(\theta)}\\
&= \frac{\sin\left(\frac{1}{1-w}\theta\right)p' + \sin\left(\frac{-w}{1-w}\theta\right)q}{\sin(\theta)}\\
&= \frac{\sin(\psi)\,p' - \sin(w\psi)\,q}{\sin[(1-w)\psi]}\\
&= \frac{\sin[(1-w)\psi]\,p + \sin(w\psi)\,q - \sin(w\psi)\,q}{\sin[(1-w)\psi]}\\
&= \frac{\sin[(1-w)\psi]\,p}{\sin[(1-w)\psi]}\\
&= p.
\end{aligned}$$

□

Lemma 2. *Given a real value $0 \le w < 1$ and any two points p and q,*

$$SLERP\left(SLERP\left(p, q, \frac{w}{2}\right), SLERP\left(q, p, \frac{w}{2}\right), \frac{w}{2w-2}\right) = p.$$

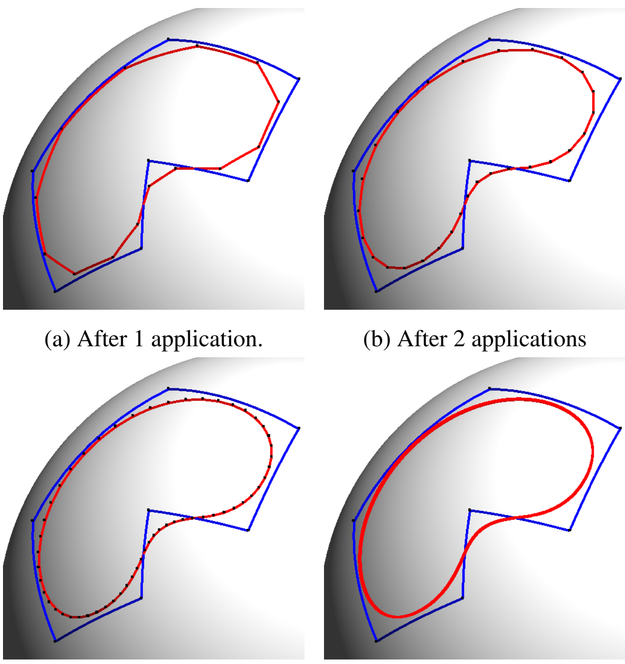
Proof. Let θ be the angle between points $p' = \text{SLERP}(p, q, \frac{w}{2})$ and $q' = \text{SLERP}(q, p, \frac{w}{2})$ and let ψ be the angle between points p and q. Notice that, by construction of SLERP, $\theta = (1-w)\psi$.

Now,

$$\begin{aligned}
&\text{SLERP}\left(p', q', \frac{w}{2w-2}\right)\\
&= \frac{\sin\left[(1-\frac{w}{2w-2})\theta\right]p' + \sin\left(\frac{w}{2w-2}\theta\right)q'}{\sin(\theta)}\\
&= \frac{\sin\left(\frac{2-w}{2(1-w)}\theta\right)p' + \sin\left(\frac{-w}{2(1-w)}\theta\right)q'}{\sin(\theta)}\\
&= \frac{\sin\left[(1-\frac{w}{2})\psi\right]p' - \sin\left(\frac{w}{2}\psi\right)q'}{\sin[(1-w)\psi]}\\
&= \frac{\sin\left[(1-\frac{w}{2})\psi\right]\left[\sin\left[(1-\frac{w}{2})\psi\right]p + \sin(\frac{w}{2}\psi)q\right]}{\sin(\psi)\sin[(1-w)\psi]}\\
&\quad - \frac{\sin\left(\frac{w}{2}\psi\right)\left[\sin\left[(1-\frac{w}{2})\psi\right]q + \sin(\frac{w}{2}\psi)p\right]}{\sin(\psi)\sin[(1-w)\psi]}\\
&= \frac{\sin^2\left[(1-\frac{w}{2})\psi\right] - \sin^2(\frac{w}{2}\psi)}{\sin(\psi)\sin[(1-w)\psi]}p\\
&= \frac{\frac{1}{2}\left[1-\cos\left[(2-w)\psi\right] - 1 + \cos(w\psi)\right]}{\frac{1}{2}\left[\cos[\psi-(1-w)\psi] - \cos[\psi+(1-w)\psi]\right]}p\\
&= \frac{\cos(w\psi) - \cos(2\psi - w\psi)}{\cos(w\psi) - \cos(2\psi + w\psi)}p\\
&= p.
\end{aligned}$$

□

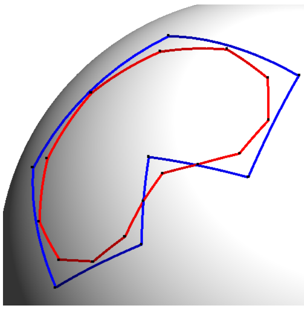
By Lemma 1, it can be seen that $\mathcal{F}_j^{-1} \circ \mathcal{F}_j(p_i) = p_i$ and $\mathcal{G}_j^{-1} \circ \mathcal{G}_j(p_i) = p_i$ for any i. Similarly, by Lemma 2, it can be seen that $\mathcal{H}_j^{-1} \circ \mathcal{H}_j(p_i) = p_i$ for any i.



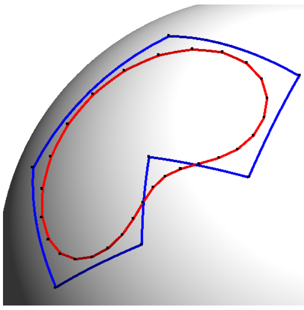
(a) After 1 application. (b) After 2 applications

(c) After 3 applications (d) After 8 applications

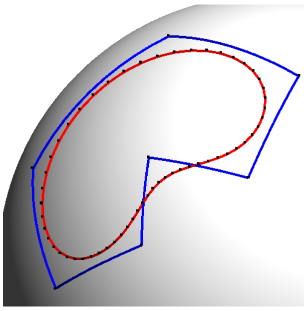
Figure 11: Results from applying our spherical subdivision scheme ($k = 2, w_0 = \frac{2}{3}, w_1 = \frac{1}{4}$) on a coarse curve (shown in blue). The subdivided curve is shown in red.



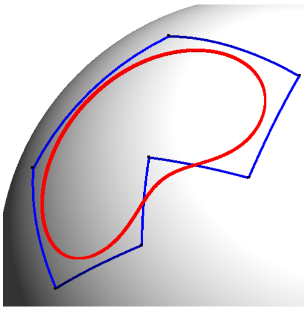
(a) After 1 application.



(b) After 2 applications

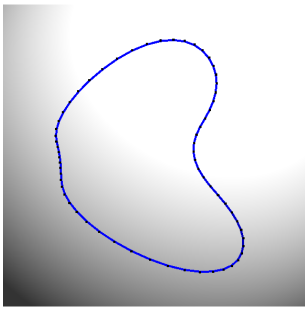


(c) After 3 applications

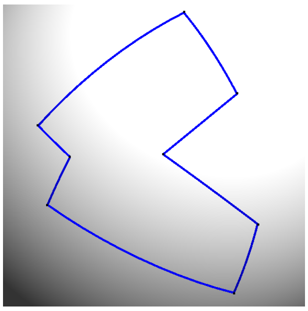


(d) After 8 applications

Figure 12: Results from applying our spherical subdivision scheme ($k = 3, w_0 = \frac{2}{3}, w_1 = \frac{1}{4}$) on a coarse curve (shown in blue). The subdivided curve is shown in red.

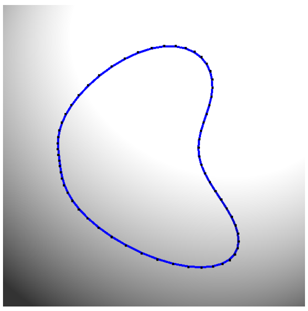


(a) Fine curve created using spherical subdivision.

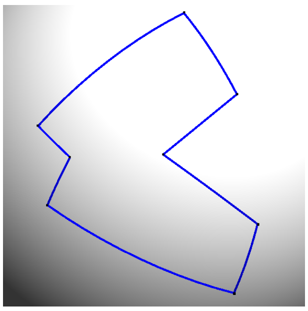


(b) The curve from (a) after 3 applications of $\mathcal{A}_{MR}$.

Figure 13: Results from applying our reverse subdivision scheme $\mathcal{A}_{MR}$ ($k = 2, w_0 = \frac{2}{3}, w_1 = \frac{1}{4}$) on a curve resulting from forward subdivision.

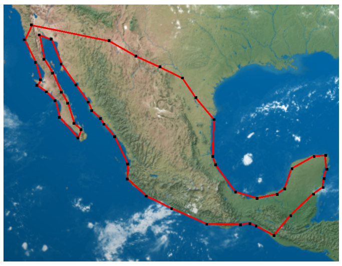


(a) Fine curve created using spherical subdivision.

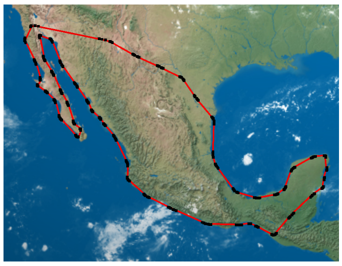


(b) The curve from (a) after 3 applications of $\mathcal{A}_{MR}$.

Figure 14: Results from applying our reverse subdivision scheme $\mathcal{A}_{MR}$ ($k = 3, w_0 = \frac{2}{3}, w_1 = \frac{1}{4}$) on a curve resulting from forward subdivision.

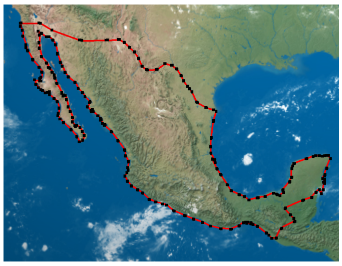


(a) The border of Mexico prepared for initial transmission by 6 applications of spherical reverse subdivision ($k = 2, w_0 = \frac{1}{10}, w_1 = \frac{1}{10}$).

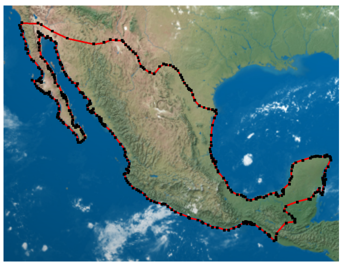


(b) Received coarse data subdivided 3 times without details.

(c) Received coarse data subdivided 3 times with the newly arrived first level of details.



(d) Received coarse data subdivided 3 times with the first level and newly arrived second level of details.



(e) Received coarse data subdivided 3 times with the first level, second level, and newly arrived third level of details.

Figure 15: Progressive transmission allows transmitted geospatial data on the sphere to be iteratively refined as the details arrive. Texture image for the Earth courtesy of www.shadedrelief.com.