Multiresolution on Spherical Curves

Alderson, Troy; Mahdavi Amiri, Ali; Samavati, Faramarz

http://hdl.handle.net/1880/50249
technical report

Downloaded from PRISM: https://prism.ucalgary.ca
Multiresolution on Spherical Curves

Troy Alderson†, Ali Mahdavi Amiri‡, Faramarz Samavati§

University of Calgary

Abstract

In this paper, we present a simple multiresolution framework 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. However, translating such combinations to spherical space is challenging. Several works perform such operations in an intermediate Euclidean space instead using some mapping (e.g. the exponential map), but such mappings cause distortions and are often complicated. We use a simple geometric construction for a multiresolution scheme on the sphere that does not require the use of an intermediate space, which is based on a modified Lane-Riesenfeld algorithm (point duplication followed by repeated averaging) that features an invertible averaging step. Such a multiresolution scheme allows one to simplify/compress and reconstruct curves on the surface of a sphere-like object — such as the Earth — simply, efficiently, and without distortion.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

1. Introduction

The sphere is a fascinating and elegant shape, and of particular interest as an approximation of the shape of the Earth. However, the surface of the sphere forms a two-dimensional non-Euclidean space which is difficult to directly operate on. For instance, we face a substantial challenge when we wish to create, edit, and manipulate curves on the surface of the sphere. Spherical curves — analogous to curves in Euclidean space — are a set of points $f_i$ on the sphere connected by geodesic lines (great circle arcs). They have been studied in the literature with applications in, e.g., vector data representation on the surface of a Digital Earth [BF01, FSP97].

Traditionally, such curves are created, edited, and manipulated in a 2D or 3D Euclidean space, then projected to the sphere. Examples of such mappings include latitude/longitude conversion, Snyder projection [Sny92], or the exponential map [dC76]. However, the use of such mappings inevitably induces distortions on the curve (see Figure 2) and can be costly to compute (particularly for mappings that aim to reduce distortion as in the case of inverse Snyder projection [HMAS12]). Hence, the ability to create, edit, and manipulate curves directly on the sphere itself is an important, albeit challenging, problem.

One of the tools that has gained prominence within the CAD/CAM industry for modeling curves and surfaces in Euclidean space is subdivision. Subdivision schemes are linear transformations that increase the resolution of a curve or surface and are often based on B-Spline knot insertion, converging to a B-Spline curve at the limit. Chaikin's corner-cutting scheme for curves [Cha74] as well as the Catmull-Clark [CC78] scheme for surfaces are some well-known examples of these methods. Such schemes can be reversed using reverse subdivision schemes.

Subdivision and reverse subdivision can be combined into a multiresolution framework with the ability to decompose a given vector of $m$ fine points $f = [f_0, \ldots, f_{m-1}]^T$ to a vector of $n$ coarse points $c = [c_0, \ldots, c_{n-1}]^T$ and associated detail vectors (i.e. wavelet coefficients) $d = [d_0, \ldots, d_{n-m-1}]^T$ [SB99, BS00]. Under this framework, the given high-resolution model (curve) can be reconstructed using the coarse points and detail vectors. A notable property of such a frame-
work 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 highly fast and efficient.

We wish to extend such a multiresolution framework to spherical curves, without leaving the spherical domain. Operations defined directly within this domain will not distort the spherical curve and will allow us to avoid the computational cost of mapping back-and-forth to an intermediate domain.

Achieving this extension is not only an interesting and fundamental research problem, but would also prove very helpful in our intended application of multiscale representations for geospatial vector data (i.e., curves representing political boundaries, road networks, geological feature outlines, etc.). The field of GIS is seeing a flood of geospatial data, and the amount and fidelity of the incoming data is set to overwhelm the current model of GIS [Pur14]. The ability to represent such data at multiple scales for compression and improved rendering and processing times is not only convenient, but essential.

One framework that is emerging to handle this data deluge is that of the Digital Earth [Goo00]. The Digital Earth framework seeks to revolutionize the mechanisms by which everyday users access geospatial data by using the curved Earth itself as a reference model. In essence, it is a digital model of the Earth and all its data, however geospatial data sets on the surface of the Earth can be extremely large and high in resolution. The ability to reduce the number of points in a spherical curve, with the ability to reconstruct the high-resolution version on the fly, would have benefits to several aspects of a Digital Earth application, such as:

1. Progressive transmission of geospatial vector data over networks, such that a simplified version of the data may be visualized or processed while details necessary to reconstruct the original data are streamed. Each simplified version of the data will remain on the sphere and will be distortion-free, and the total amount of bandwidth needed would be equal to the bandwidth needed to send the high-resolution version.
2. Level-of-detail control for rendering and visualization. High-resolution versions of the data may be reconstructed without re-streaming the data or saving a high-resolution copy.
3. Fast estimates for geospatial queries (e.g., intersection of two regions, buffer regions) using simplified data, that may then be refined by adding details back into the simplified data.

The restriction to operate directly within spherical space presents the crux of the problem. While two-point interpolations on the sphere can be computed using spherical linear interpolation (SLERP), weighted averages of more than two points (by which subdivision schemes are typically understood and implemented) are not uniquely defined in spherical space. Hence, a multiresolution scheme defined in spherical space must be constructable using well-defined sequences of two-point interpolations. This holds for all its constituent operations: subdivision, reverse subdivision, detail computation (i.e., decomposition), and detail restoration (i.e., reconstruction).

In this paper, we introduce a construction for multiresolution and smooth multiresolution in spherical space that satisfies these constraints, allowing points to be combined without ambiguity, and which operates directly on the sphere without distortion. The construction is based on a modified version of the Lane-Riesenfeld algorithm [LR80], which uses two simple geometric operations: point duplication and midpoint finding.

2. Related Work

Curves that lie on surfaces (including manifolds, triangle meshes, and spheres) have been well studied in the literature [ANS96, FSP97, BF01]. 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 [PH03, GCP*10]. Spherical curves are particularly of interest within the Digital Earth framework [WKW*03, Goo00], which represents the Earth as a curved surface rather than as a flattened map.

Multiresolution for curves and surfaces is a well studied subject [Gar99, SDS96]. 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 [Cas12, SB99, BS00]. In such a multiresolution framework, 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 weighted averages. Weighted averages in Euclidean space are very useful, as they can be efficiently used for texture mapping, sampling, and smoothing. As a result, redefining weighted averages within manifold, spherical, and Riemannian spaces has been studied in several previous works [PBDS13, Rus10].

Weighted averages on the sphere have been approached via least squares optimization [BF01]. This approach becomes inefficient when we are dealing with a large number of spherical points, which is the case for geospacial vector data of the type encountered in Digital Earth frameworks. In addition, 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 based on this method.

An alternative approach is to compute the average in an intermediate space and then project the result back to the sphere [ANS96, LBS06], but this causes distortions to the spherical curve which are undesirable. Furthermore, the mapping itself could potentially be a costly operation.

Subdivision for curves on general manifolds has been proposed in [ESHMMMV09, MMVC08, WD05, WP06] and for spheres in particular in [Ode08, SG05]. However, these works do not present corresponding spherical reverse subdivision/multiresolution schemes. Similarly, the well-known Ramer-Douglas-Peucker algorithm [DP73] can be used to reduce the number of points in a curve, but is a simple downsampling and does not support lossless reconstruction of the original curve.

In [GW12], 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, but note that it is not clear how to achieve perfect reconstruction in the general case. Our multiresolution scheme is neither interpolating nor midpoint-interpolating, but achieves perfect reconstruction.

Multiresolution for spherical domains has also been proposed in wavelet form [SS95, LF08]. 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. The Lane-Riesenfeld Algorithm

Instead of weighted averages, it is possible to implement subdivision using simpler geometric operations that are easier to translate into other spaces. In particular, there exists a simple construction for B-Spline subdivision schemes of arbitrary degree based on repeated averaging that uses only midpoint operations, known as the Lane-Riesenfeld algorithm [LR80]. The generalization of the Lane-Riesenfeld algorithm to spherical space has been performed in [SG05], which we reiterate here, before extending the algorithm to a spherical multiresolution framework.

The construction for a B-Spline subdivision scheme of degree k 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 the geodesic midpoint of the vertex and its consecutive neighbour.

Let P be the subdivision transformation for the desired subdivision scheme. Then, \( f = P(c) \) and \( P = S^k \circ D \) where \( D \) is duplication transformation (in fact, a Haar subdivision operation) and \( S \) is the averaging transformation. Given two consecutive points \( p_0 \) and \( p_1 \), the geodesic midpoint of these two points can be found via \( \text{SLERP}(p_0, p_1, \frac{1}{k}) \), or by normalizing the Euclidean midpoint \( \frac{1}{2}p_0 + \frac{1}{2}p_1 \) to the sphere.

Note, of course, that the subdivision is only valid if the angle between consecutive 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°, special care must be taken to correctly compute the spherical midpoint. However, this case is rarely of interest.

This “spherical Lane-Riesenfeld” operation, which is defined for closed curves, can be adapted to open curves by first applying the subdivision, then discarding the last k points of the resulting curve, and finally replacing the endpoint positions of the resulting curve with the endpoint positions of the original curve.

4. Invertible Averaging

While this generalization of Lane-Riesenfeld allows us to achieve forward subdivision on the surface of the sphere, it unfortunately does not allow for reverse subdivision (which is necessary for a full multiresolution), as the averaging step is not invertible. One possibility is to use optimization, however this would present the challenge of determining a method for efficiently evaluating a compact set of details for lossless reconstruction. Furthermore, keeping the vertices on
4.1. Spherical Subdivision and Reverse Subdivision

In this section, we present a subdivision and reverse subdivision scheme that operate directly on the surface of a sphere. The subdivision scheme is similar to the spherical Lane-Riesenfeld algorithm, using the duplication step \( D \) followed by \( k \) iterations of an averaging step. 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, we redefine the subdivision transformation \( P \) such that the averaging step \( S \) is replaced by an invertible averaging step \( F \), composed of two invertible operations, \( F_1 \) and \( F_2 \), i.e. \( F = F_2 \circ F_1 \).

As demonstrated in Figure 3, \( F_1 \) fixes every other vertex (say, vertices with odd indices) and moves each unfixed vertex (those with even indices) to its geodesic midpoint with its consecutive neighbour. \( F_2 \) is similar but moves vertices with odd indices and fixes those with even indices.

A property of \( F_1 \) and \( F_2 \) is that they have inverses \( F_1^{-1} \) and \( F_2^{-1} \), hence \( F^{-1} = F_1^{-1} \circ F_2^{-1} \). The application of \( F_1^{-1} \) moves each vertex with an even index such that the arc length is doubled between it and its consecution neighbour. That is, given points \( p_0 \) and \( p_1 \), \( F_1^{-1} \) undoes the effect of \( F_1 \) by taking \( p_0' = \text{SLERP}(p_0, p_1, -1) \). An equivalent calculation is to reflect \( p_1 \) about the vector between \( p_0 \) and the sphere’s center, i.e.

\[
p_0' = 2 \frac{p_0 \cdot p_1 - p_0^0}{p_0 \cdot p_0^0} p_0 - p_1.
\]

\( F_2^{-1} \) does the same thing for vertices with odd indices. See Figure 4 for an illustration.

While the duplication transformation \( D \) does not have an inverse, it is the subdivision operation behind Haar wavelets, whose reverse subdivision scheme replaces each pair of vertices by their midpoint [SDS96], i.e. \( c_i = \frac{1}{2} f_{2i} + \frac{1}{2} f_{2i+1} \). In spherical space, this operation replaces each pair of vertices with their geodesic midpoint, and we will denote this reverse Haar transformation by \( M \). Notice that \( M \) is a downsampling operation, unlike the fairing operations \( S \) and \( F \).

Hence, we define a spherical subdivision transformation \( f = P(c) \) where \( P = (F_2 \circ F_1)^k \circ D \), with reverse transformation \( c = A(f) \) where \( A = M \circ (F_1^{-1} \circ F_2^{-1})^k \). 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 will introduce \( c_i \) and \( c_{i+1} \) for some \( i \) with an angle \( \theta \geq 180° \) between them, which invalidates the forward subdivision operation.

5. Spherical Multiresolution

Since \( F_1 \) and \( F_2 \) are perfectly invertible, high resolution details of the curve are only lost during the application of the reverse Haar operation \( M \). Hence, it is possible to achieve multiresolution in this case using the foundations of Haar wavelets.

In Euclidean space, the Haar detail vectors are found by taking half the vector between each pair of vertices [SDS96], i.e. \( d_i = \frac{1}{2} f_{2i} - \frac{1}{2} f_{2i+1} \). In spherical space, we can generalize...
this so that the details are half the rotation between each pair of vertices. We denote the transformation that returns these detail rotations as \( B \).

Reconstructing the original curve after Haar subdivision, which duplicates the vertices, involves taking the detail vector corresponding to a vertex, adding it to the first copy, and then subtracting it from the second, i.e. \( f_{2i} = c_i + d_i \) and \( f_{2i+1} = c_i - d_i \). In spherical space, after duplication we take the detail rotation corresponding to a vertex, apply the positive rotation to the first copy and the negative rotation to the second. Denote this transformation by \( Q \).

Now we may define our multiresolution scheme on spherical curves. The reconstruction step determines the fine points \( f = (F_2 \circ F_1)^k \circ Q(D(c), d) \), and the decomposition step determines the coarse points \( c = M \circ (F_1^{-1} \circ F_2^{-1})^k(f) \) and detail rotations \( d = Q \circ (F_1^{-1} \circ F_2^{-1})^k(f) \). 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 over-representation.

### 6. Smooth Spherical Multiresolution

Reverse subdivision tends to exaggerate the shape of the simplified curve to offset the smoothing effects of subdivision. Reducing these shape exaggerations within a multiresolution framework so that the simplified curve better resembles the original is the goal of a smooth multiresolution framework [SS13]. We generalize this smooth multiresolution approach to spherical space, making use of the modified (invertable) Laplacian operator.

Regular Laplacian smoothing moves each vertex \( c_i \) of a curve to the midpoint of its neighbour vertices (i.e. \( c_i' = \frac{1}{2}(c_{i-1} + c_{i+1}) \)), but the operation is not invertible. In order to make this operation invertible, the Laplacian operator can be modified so that every other vertex is fixed at a time, and the unfixed vertices are moved towards, but not placed at, the midpoint of each neighbour’s neighbours. That is, \( c_i' = (1-w)c_i + w\left(\frac{1}{2}c_{i-1} + \frac{1}{2}c_{i+1}\right) \), where \( 0 \leq w < 1 \) is a weighting parameter.

In spherical space (see Figure 5), let \( G = G_2 \circ G_1 \) be the smoothing operation that generalizes the modified Laplacian. \( G_1 \) moves the vertices with even indices to \( c_i' = \text{SLERP}(c_i, \text{SLERP}(c_{i-1}, c_{i+1}, \frac{1}{2}), w) \), and \( G_2 \) moves the vertices with odd indices.

The inverse of the modified Laplacian fixes every other vertex at a time, finds the midpoint between a vertex’s neighbours, and calculates the vertex’s original position using the vector between the midpoint and the

![Figure 5: Illustration of the modified Laplacian and its inverse on the sphere.](image)

smoothed position. In spherical space, \( G^{-1} = G_1^{-1} \circ G_2^{-1} \), where \( G_1^{-1} \) moves the vertices with even indices to \( c_i' = \text{SLERP}(c_i, \text{SLERP}(c_{i-1}, c_{i+1}, \frac{1}{2}), w) \), and \( G_2^{-1} \) moves the vertices with odd indices.

We apply the smoothing step \( G \) after reverse subdivision, such that \( c = G \circ M \circ (F^{-1})^k(f) \), and apply the inverse smoothing step before the duplication during the reconstruction, such that \( f = F^k \circ \tilde{Q}(G^{-1} \circ D(c), d) \).

### 7. Results

Result images from applying our multiresolution scheme on spherical curves can be found on page 7.

Figure 6 shows some result images after applying three applications of subdivision on a coarse spherical curve. Smoother subdivisions can be achieved by increasing \( k \), and in all cases, the curves remain on the sphere.

A demonstration of our reverse subdivision technique is shown in Figure 7. A spherical curve is shown after two applications of both smooth and non-smooth reverse subdivision. Our smooth schemes use a weighting parameter of \( w = \frac{1}{2} \), which makes it possible to compute and invert the Laplacian smoothing using the midpoint and reflection operations that form the multiresolution scheme.

In Figure 8, we illustrate the iterative refinement of real geospatial vector data representing nation boundaries under progressive transmission. A coarse (i.e. reverse subdivided) version of the data may be transmitted initially and quickly over network, and is enhanced as multiresolution details for three successive resolutions arrive via transmission.

### 8. Conclusions and Future Work

We have presented a new multiresolution on spherical curves. The key behind our construction is the use of simple geometric transformations (i.e. duplication, averaging, and reflection) that are generalizable to the spherical domain and make for efficient decomposition and reconstruction operations. We additionally present a construction for smooth multiresolution on the sphere by generalizing Laplacian smoothing.

As a potential direction for future work, it would be interesting to extend these schemes to other manifolds, particularly ellipsoids and geoids. Determining those manifolds
with simple midpoint operations could be an important first step. Also, the geometric properties of the proposed methods remain to be found, although we suspect that, given the degree \( \deg \), the continuity of the forward subdivision should be \( C^{\deg -1} \).

References


Figure 6: Three applications of our subdivision scheme on a spherical curve. Vertices of the spherical curves are marked with small black boxes connected by geodesic lines. The original curve is shown in blue, whereas the generated curve is shown in red.

Figure 7: Two applications of reverse subdivision on a spherical curve.

Figure 8: Progressive transmission allows transmitted geospatial data on the sphere to be iteratively refined as the details arrive. To improve the quality of the results, we do not apply the inverse smoothing operation $G^{-1}$ during subdivision steps in which the details are not known. Texture image for the Earth courtesy of www.shadedrelief.com.