Continuous Indexing of Hierarchical Subdivisions of the Globe

John J. Bartholdi, III [*] Paul Goldsman

December 8, 2000

Abstract

We describe how to create a continuous global index of the surface of the earth. The index is based on a hierarchical subdivision of the surface into triangular regions in which each region is assigned a numerical label according to a spacefilling curve. Sequential labels are assigned to adjacent regions, so labels can be sorted to create a continuous one-dimensional index. Benefits of continuity include the implicit preservation of adjacency information, and the ability to vary resolution at different locations.

Previously suggested schemes based on similar models produce indices that are discontinuous. Unfortunately, discontinuities degrade the usefulness of an index, as we show by comparing continuous and discontinuous schemes based on performance criteria such as the ability to preserve spatial adjacency. The best index appears to be the continuous one based on the Sierpinski spacefilling curve.

Key words: *geographic information system, spacefilling curve, triangulation, global index*

[*] School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0205 USA. E-mail: john.bartholdi@isye.gatech.edu, pgoldsman@hotmail.com.

1 Introduction

We describe how to construct a continuous index for geographic information on the globe. Key features of the model include: 1. It is based on the hierarchical subdivision of the surface of the earth into triangles. 2. Subregions are ordered in a continuous way, so that sequential index values are given to adjacent subregions. 3. Subregions at a given level of the hierarchical subdivision are approximately equivalent (in size and shape) and correspond to equal-length ranges of index values.

In one common model of the globe, the surface is initially divided into a spherical octahedron (the spherical equivalent of an octahedron, where the surface of the sphere is divided into eight equivalent spherical triangles). The base triangles are then hierarchically subdivided as needed to capture local surface detail. In other words, the subdivision is deeper where more detail is required. Our research is concerned with finding continuous orderings of such triangle coverings of the sphere.

We review some proposed discontinuous indexing schemes for similar models. We then show how to create a continuous index of the sphere, based on a spherical spacefilling curve, and provide procedures for converting between coordinates and position on the curve, and related operations of finding ancestors, descendants, and neighbors of cells in the subdivision.

1.1 Relevance of our research

Our research describes a continuous index of the surface of the earth based on a hierarchical subdivision, though the index could also be used to index spatial data structured according to a different model. This scheme has a number of useful features.

Hierarchical organization. Hierarchical models have the inherent ability to reduce data size, and vary resolution at different locations. This is useful since the density of features of interest on the surface of the earth is not uniform; such models conserve space by increasing resolution in dense areas, and decreasing it in sparse areas.

Weibel and Dutton (1999) discuss hierarchical tessellations in the context of GIS data generalization problems. An example of such a problem is the need to increase or decrease the level of spatial detail displayed in a map, when zooming in or out, respectively. This problem is sometimes handled in GIS by storing multiple versions of maps at varied levels of detail.

Continuous ordering. In our model, each cell in the subdivision receives a unique numerical label. These labels are assigned in a spatially continuous way: Cells with consecutive labels are spatially adjacent, and spatially near cells tend to have close labels. This is useful for operations that depend on clustering and spatial adjacency in general. Spatial range searching is one example (Nulty 1993). Clustering is also important when large data sets are stored on slower secondary storage devices such as hard disks. A basic principle of spatial data is that spatially near points tend to be more related than distant points. Such points will more often be accessed together, so one can reduce the number of disk accesses by physically storing them near each other. We ensure that nearby points tend to have nearly equal indices by ensuring that adjacent cells tend to have adjacent ranges of index.

We consider cells to be adjacent if they have a common vertex. In the literature, edge adjacency is often required (for example, between cells of rectangular grids (Abel and Mark 1990)). Our slightly weaker form of adjacency allows us to get much stronger results.

Equivalent subdivision. We show how to create a subdivision in which cells at a given level of the hierarchy tend to be approximately equivalent (in size and shape). As we shall see, the number of digits in the label of a cell equals its depth in the hierarchy. The length of a cell label also gives an indication of the size of the cell, and cells with labels of the same length tend be approximately equivalent. Thus, label length can be thought of as a rough proxy for data precision (Dutton 1989, 1996).

2 Hierarchical Subdivision of the Sphere

A number of researchers have developed hierarchical data models to represent the surface of the earth. These models partition the surface of the earth into spherical triangles, and then hierarchically subdivide these triangles to increase detail at different areas on the earth.

In practice, most researchers find it more convenient to work with planes rather than spherical surfaces. In their models, the earth is represented as a triangular-sided polyhedron inscribed within a sphere. The faces of the polyhedron are projected onto the surface of the sphere, to make a correspondence between the points on the polyhedron and the points on the sphere. Increasing levels of detail are achieved by hierarchically subdividing the triangular faces of the base polyhedron, and projecting these onto the sphere, until sufficiently small regions can be distinguished. In other words, triangular subregions on a side of the polyhedron correspond to spherical triangular regions on the sphere. Working directly with the sphere, or with the corresponding polyhedron, is largely an implementation issue, the underlying ideas are the same. However, speaking in terms of sides of polyhedrons often results in plainer language ("triangle" versus "spherical triangle"), so we will in most cases adopt that convention.

Suppose, for example, that the model is based upon an inscribed octahedron. The octahedron is projected onto the sphere, creating a spherical octahedron, and subdivided to create a triangular covering of the earth. Such hierarchical structures can be subdivided to different depths at different locations on the globe, to account for differing density of features of interest; for background, see Laurini and Thompson (1992, pp. 244–246). The authors we discuss work with either the octahedron or the icosahedron.

An important aspect of such subdivisions is a scheme for assigning labels to the triangular cells. Such labels uniquely identify cells and, by extension, points within cells. A point on the surface of the globe inherits the label of its enclosing cell. Labels can be sorted to create a one-dimensional spatial index of the global surface.

A *labeling* of a subdivision is an assignment of unique codes (or "labels") to each cell of the subdivision. A *continuous spatial index* (or *ordering*) is one in which cells with sequential index values are at least vertex-adjacent. Indexing schemes thus far proposed for hierarchical global subdivisions have been discontinuous.

Our discussion will focus on this aspect of global subdivisions. We review a number of proposed discontinuous indexing schemes for hierarchical subdivisions of the globe, and suggest two new indexing schemes that are continuous.

2.1 Spherical subdivisions

Hierarchical subdivisions of the sphere are in some ways the spherical equivalent of planar quadtrees, and they share many of the advantages of such hierarchical data models—in data representation, storage and access, for example (Dutton 1989, 1996, Mark and Lauzon 1984, Samet 1989, 1990).

Hierarchical structures have mainly been discussed on the plane, but not exclusively. For examples of hierarchical structures used to model triangulated irregular networks, see Barrera (1989), Scarlatos and Pavlidis (1992), Sotomayor (1978). Hierarchical subdivisions of the earth based on quadrilaterals have also been proposed. Mark and Lauzon (1985) describe some attempts, as well as their own quadtree-based approach. In some ways, global subdivisions with quadrilateral faces are simpler to work with than those with triangular faces. Quadrilateral shapes are more similar to the way we are accustomed to thinking about geography and maps, and edges may map to parallels or meridians. But it is difficult with these structures to approach equivalency in size and shape among cells at a given level in the hierarchical subdivision. For example, models based on latitude and longitude suffer from the fact that as we approach the poles, cells become smaller and more triangular.

Important aspects of a hierarchical subdivision of the sphere based on triangles are: 1. The initial division of the spherical surface into triangular domains. 2. The process of subdividing the domains.

3. The process of labeling cells. One would prefer a hierarchical subdivision of the sphere in which all cells at a given level of the subdivision were equivalent in size and shape (apart from reflection). Unfortunately, the only known equivalent hierarchical subdivisions of the sphere are those suggested by the way citrus fruits divide naturally into "lunes." In the citrus-fruit family of subdivisions, cells become thinner and thinner, converging in the limit to arcs of great circles (*geodesics*), rather than to points, and for this reason, such subdivisions are unlikely to be useful. Apart from these examples, there is no known way to hierarchically subdivide the sphere into more than 120 equivalent domains. The spherical icosidodecahedron (120 equivalent triangles) is obtained by dividing each face of a spherical icosahedron (twenty triangles) into six congruent scalene spherical triangles by bisecting each angle (Croft, Falconer and Guy 1991, p. 90, Davies 1967, Wenninger 1979).

Therefore, the construction of a hierarchical global data structure is in part an exercise in compromise. The cells at a given level will differ in size or shape or both (White, Kimerling, Sahr and Song 1998, for example), and perfect adjacency-preservation is also unobtainable (Sagan 1994).

2.2 Initial partition of the sphere

The Platonic solids are reasonable starting points for a spherical subdivision. Of the five Platonic solids, three have triangular faces: the tetrahedron (four faces), the octahedron (eight faces), and the icosahedron (twenty faces). The other Platonic solids are the cube (six faces) and the pentagonal dodecahedron (twelve faces). The icosahedron has the greatest number of initial faces, and would therefore show the least distortion in the subdivision, but practically speaking the larger number of faces makes it somewhat harder to work with. The octahedron has more distortion, but has the advantage that its faces and vertices map to important global features—meridians, the equator, and the poles (Goodchild and Shiren 1989).

The triangles of the initial partition need not be equilateral. Distortion could be decreased considerably by dividing each equilateral triangular side of an initial Platonic figure into equivalent scalene triangles, to start with a partition of the surface with a much greater number of equivalent base triangles. A spherical equilateral triangle can be divided into a maximum of six equivalent spherical scalene triangles. Therefore, a solution with minimal distortion could start with up to 120 equivalent domains by beginning with an icosahedron and dividing each of the equilateral triangular faces into six equivalent scalene triangles.

We discuss regular partitions, but regularity is not necessary. The initial partition of the earth into spherical triangles could take other considerations into account, such as specific earth features: land

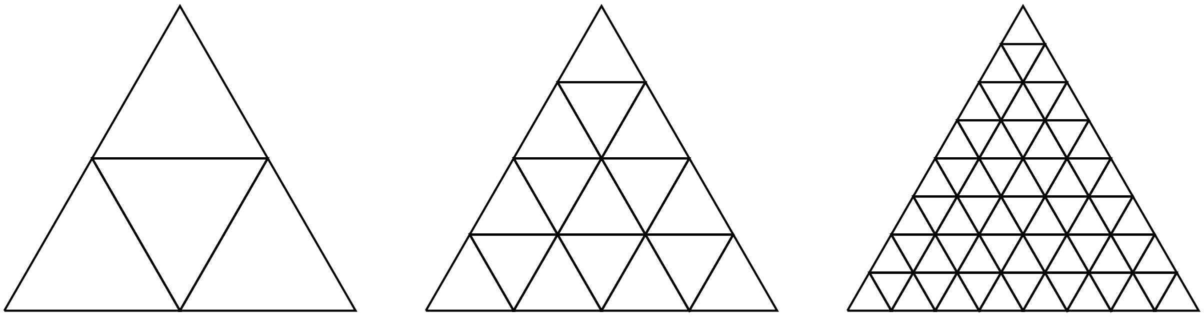

Figure 1: Quaternary subdivision

masses, weather patterns, political subdivisions, or mineral deposits, for example.

2.3 Subdivision of triangular cells

There are a number of ways to hierarchically subdivide an equilateral triangle. All of these are subject to distortion when transferred to the surface of the sphere. In other words, all eventually result in non-equivalent subtriangles. Different decisions will have different effects on the uniformity of shape and size of cells within a given level of the hierarchy, as well as on the ease of calculation. The authors we discuss favor the subdivision shown in Figure 1, in which a triangle is subdivided by joining the midpoints of each side with a new edge, to create four equivalent (on the plane) equilateral subtriangles. We will refer to this as the *quaternary subdivision*.

The quaternary subdivision is a good compromise. It is relatively easy to work with, and non-distorting on the plane—a planar equilateral triangle is divided into four equivalent equilateral triangles. But it is distorting on the sphere, resulting in three non-equilateral, but equivalent, side triangles, and one larger equilateral central triangle.

We show how to create a continuous index based on the quaternary subdivision. We also show how to create a continuous index based on a *binary subdivision* that is somewhat simpler to work with. In this subdivision, a triangle is subdivided by adding a new edge joining one of the vertices with the midpoint of its opposing side, to create two subtriangles (Figure 2).

Starting from a partition of the sphere based on the cube, it is possible to make up to three non-distorting binary subdivisions, resulting in a total of 48 equivalent domains on the sphere: The first subdivision divides each of the six spherical squares into two equivalent spherical isosceles triangles; the second and third subdivisions divide each of the triangles. In contrast, as we have seen, the very

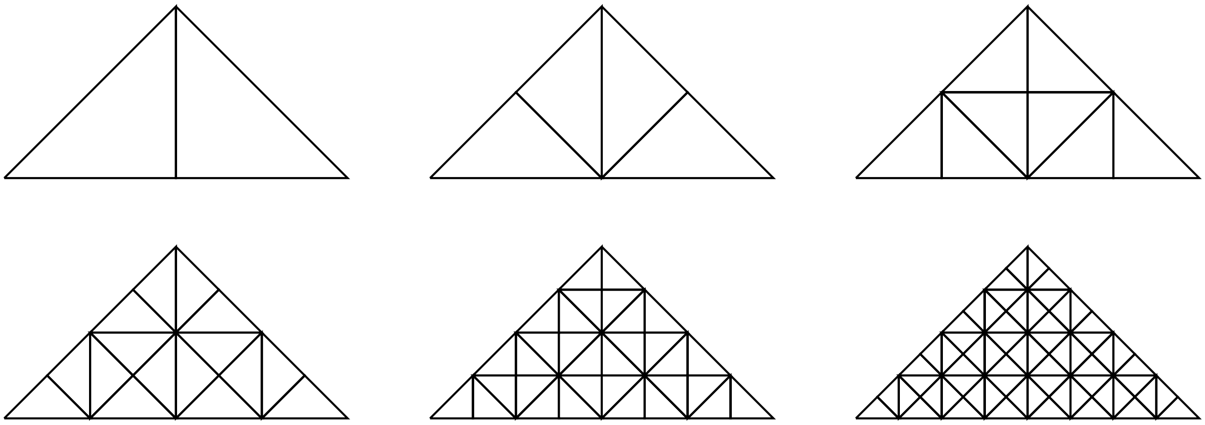

Figure 2: Binary subdivision

first quaternary subdivision of a spherical equilateral triangle is distorting. The quaternary subdivision, based on the usual octahedron, for example, never exceeds its initial eight equivalent domains. In this case, the binary subdivision would result in considerably less cell distortion.

Another partition often discussed is that of an equilateral triangle into nine equivalent equilateral subtriangles (White et al. 1998). Though we do not address this explicitly, our methods also apply.

2.4 Labeling and ordering hierarchical subdivisions

All of the labeling schemes we discuss, apart from one variation, construct the label of a cell in the same basic recursive way. First, each side of the base polyhedron is assigned a digit. Then the label of a subcell is created by right-appending a digit to the label of its parent cell. Where the labeling schemes differ is in how that digit is chosen.

This continues at each stage of the subdivision. For example, suppose we are starting with an octahedron and using the quaternary subdivision. Then each of the eight base triangles is subdivided into four subtriangles, labeled: 0, 1, 2, and 3. (We ignore any digit identifying the base triangle.) Each of these triangles may be further subdivided. Triangle 1 would subdivide into triangles 10, 11, 12, and 13; triangle 2 would subdivide into triangles 20, 21, 22, and 23; triangle 22 would subdivide into triangles 220, 221, 222, and 223, etc. This continues at each level, until "sufficient precision" is reached.

The result is a tessellation of the sphere into triangular cells. Each cell has a unique label. Each label is a string of digits that can be treated as a base 4 integer, or in the case of a binary subdivision, a base 2 integer. The labels can be sorted to create a spatial index of the surface.

2.5 Desirable characteristics of spatial orderings

A number of authors have discussed useful characteristics of spatial orderings (Abel and Mark 1990, Goodchild 1989a, 1989b, Goodchild and Grandfield 1983, Laurini and Thompson 1992, Nulty 1993, Samet 1995). We briefly review some of the issues.

One of the most important characteristics of a spatial ordering is the preservation of adjacency relationships: It is desirable that points that are near in space have similar labels, and vice versa. This is useful in a variety of common spatial and geographical operations, such as neighbor-finding, map-drawing, spatial searches, range queries, and other operations that depend on good clustering characteristics. Ideally, cells would be adjacent if and only if their labels were consecutive, but this is not possible in general.

As many applications involve paths or circuits, it is often useful if an ordering constitutes a reasonable path or a circuit through a space. For example, the Sierpinski curve (Sierpinski 1912) leads to a reasonable heuristic for the Traveling Salesman Problem. Bartholdi and Platzman (1988) review this and related results. Other things being equal, short, continuous, smooth paths are desirable.

Certain orderings are easier to work with than others. The most basic computational operations are converting between index order and spatial position, but for any practical application, there are a host of other operations to be implemented. These may include operations related to handling spatial regions, such as finding neighbors, ancestors, and descendants, as well as edge-finding, range queries, and conversion between latitude and longitude.

An ordering is said to be *stable* if the relative order of individual points does not change when the domain (that is, the space) is extended. This is less important in global data models, since the size of the space is fixed (Nulty 1993, Samet 1995). A more useful kind of stability for our purposes relates to subdivision level. The order of points on the surface should remain the same at different levels of the subdivision. We shall refer to this property as *order-consistency*; it is similar *mutatis mutandis* to Abel and Mark's (1990) concept of a *quadrant-recursive* ordering.

Nulty (1993) points out that symmetric orderings often have other desirable characteristics, such as simpler implementations (in other words, better usability), and in some cases better clustering and routing performance.

Continuity, like symmetry, is not an end in itself, but tends to imply other desirable characteristics. For example, in a continuous ordering based on the Sierpinski or Hilbert curve (Hilbert 1891), adjacency is preserved in the sense that adjacent points in the ordering are assigned to adjacent domains, and adjacent domains tend to be near each other in the ordering. A continuous ordering of cells also implies

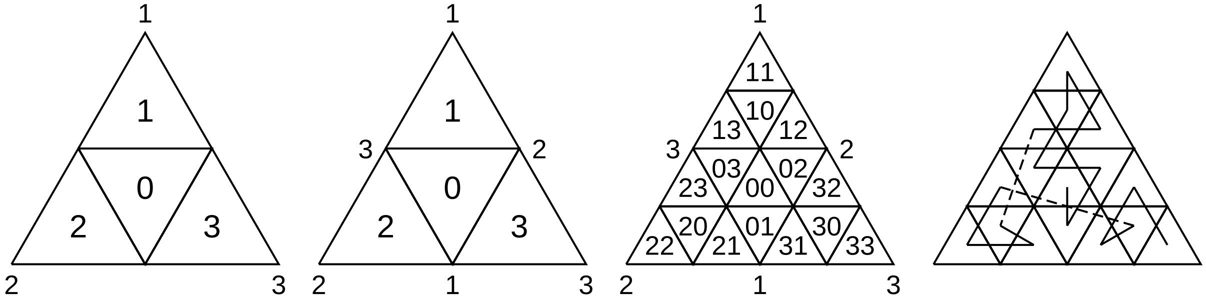

Figure 3: Dutton's Quaternary Triangular Mesh (QTM)

a path or a circuit, and all things being equal, such paths are likely to be shorter than in discontinuous orderings.

3 Literature Review

We discuss four proposed orderings of the sphere based on the hierarchical subdivision of triangles, those of: Dutton (1984, 1989, 1990, 1996), Goodchild and Shiren (Goodchild 1989b, Goodchild and Shiren 1989, 1990, 1992), Fekete and Davis (Fekete 1990, Fekete and Davis 1984), and Otoo and Zhu (1993). Among the four approaches, we find three distinct labeling schemes, none of which is continuous. We follow by suggesting two new schemes for continuous labeling.

3.1 Dutton's ordering

Dutton's *Quaternary Triangular Mesh* (QTM) is based on the hierarchical subdivision of an octahedron inscribed within a sphere. One of the advantages of the octahedron is that it can be aligned on the globe in such a way that its vertices and edges line up with important global features such as the poles, the equator, and meridians.

Figure 3 shows the first two levels of subdivision of a triangle labeled according to Dutton's QTM. Each triangle is divided into four subtriangles: a central triangle, a triangle pointing upward or downward, a left triangle, and a right triangle. Each subtriangle is assigned a digit between '0' and '3', as shown, and the label of the subtriangle is created by right-appending its digit to the label of its parent. This makes for simple ancestor-finding and descendant-finding operations.

An interesting element of Dutton's labeling is that cells surrounding certain vertices all have the same

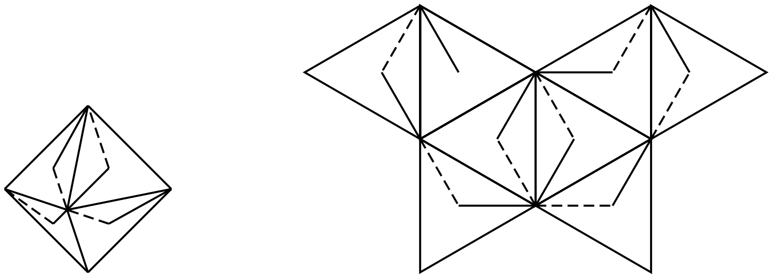

Figure 4: Dutton ordering over an octahedral surface

terminating digits. He refers to these vertices as *attractors*, and they are of some use in determining adjacencies, and other spatial operations.

Dutton's ordering is discontinuous, but adjacency-preserving to some extent: Points near in space tend to be near in the ordering, and vice-versa. A path based on the ordering would have many discontinuities (dashed lines in Figure 3). Since one end of the path is at the center of the triangular space, and the other end is at a vertex, a circuit or path based on the ordering would have additional large discontinuities between each of the initial triangles in the partition of the surface of the sphere. Figure 4 shows such a path over an octahedral surface. The ordering is order-consistent across levels of the subdivision. Many of these comments apply equally to the orderings of Goodchild and Shiren (Section 3.2), and Fekete and Davis (Section 3.3).

3.2 Goodchild and Shiren's ordering

Goodchild and Shiren developed a variation of Dutton's QTM also based on octahedrons (Figure 5). The label of a subtriangle is likewise created by appending a digit to the label of its parent, so ancestor and descendant-finding are simple. The major change from Dutton is in simplifying the labeling scheme. Like Dutton, the right-most digit of the central cell of a subdivided triangle is always '0'. But now, the up or down triangle is always '1'; the right triangle is always '2', and the left triangle is always '3'. This scheme simplifies conversions between latitude and longitude, and coordinates, as well as certain spatial algorithms.

Goodchild and Shiren's ordering has many of the same characteristics as Dutton's. It is discontinuous, but adjacency-preserving to some extent. It is order-consistent, but paths and circuits based on the ordering are very discontinuous.

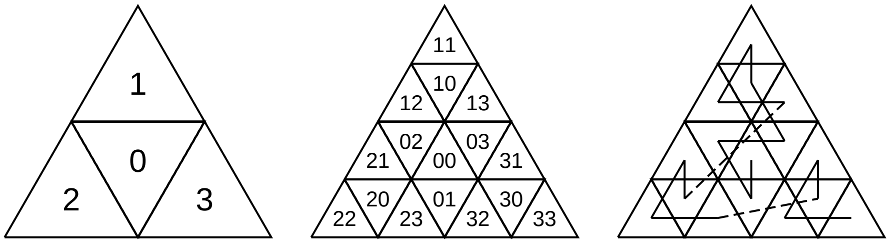

Figure 5: Goodchild and Shiren's version of the QTM

3.3 Fekete and Davis's ordering

Fekete and Davis's *spherical quadtree* (SQT) is based on an inscribed icosahedron. The cell sequencing is the same as Dutton's, except reversed, and the labels themselves are slightly different in form. The ordering is discontinuous, but still adjacency-preserving to some extent, and order-consistent across levels.

Starting with the twenty-sided icosahedron reduces the differences in size and shape of cells in the subdivision. Cells are therefore much more regular than in Dutton's or Goodchild and Shiren's schemes. However, most edges and vertices of the base icosahedron cannot be aligned with useful global references. In addition, due to the larger number of initial faces, computation is more involved.

3.4 Otoo and Zhu's ordering

Otoo and Zhu's *semi-quadcode* (SQC) ordering is based on an inscribed octahedron. This interesting scheme starts with a quadtree-type labeling. (The quadtree is a hierarchical subdivision of the square.) It then divides each quadtree cell into two triangles, and uses just half of the resulting cells—hence, a 'semi' quadcode. Figure 6 shows how quadtree labels are derived and divided into semi-quadcodes. Figure 7 shows how the quadtree labels form the basis of an ordering of the quaternary subdivision.

The SQC labeling simplifies neighbor-finding and certain other spatial operations. It has the disadvantage that the spatial ordering of certain regions may change at different levels of the subdivision. That is, it is not order-consistent. For example, cell 01 precedes 02 at Level 1 of the subdivision (not shown). But at Level 2, 012, a child of 02, now precedes 021, a child of 01 (Figure 7).

A related disadvantage is a more complicated relationship between the labels of ancestors and descendants. In the other labeling schemes, one merely had to right-remove digits from the label of a cell to generate a list of its ancestors. Similarly, the descendants of a cell could be generated by right-appending

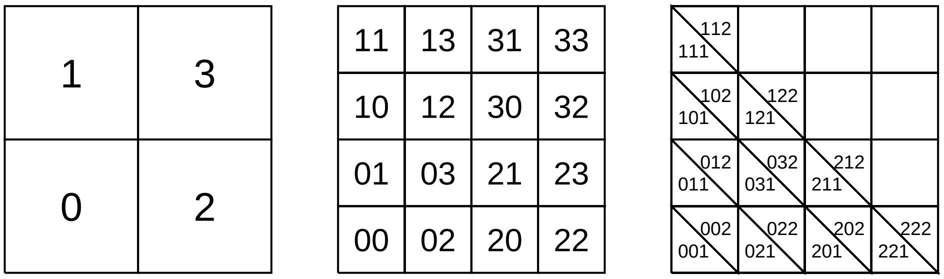

Figure 6: Derivation of semi-quadcodes (SQC) from the quadtree

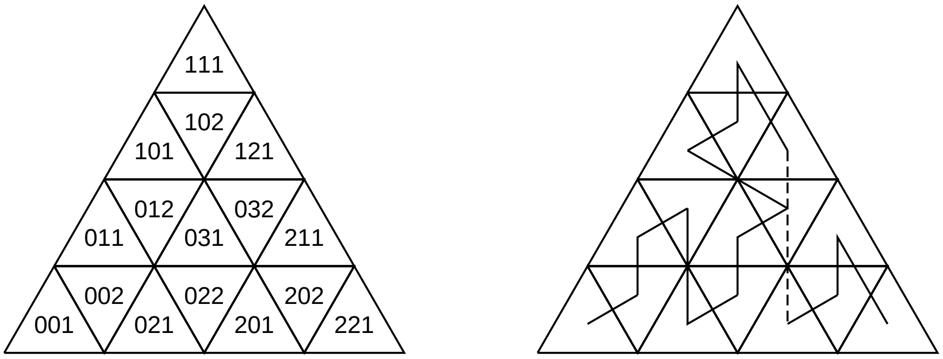

Figure 7: Otoo and Zhu's semi-quadcodes (SQC)

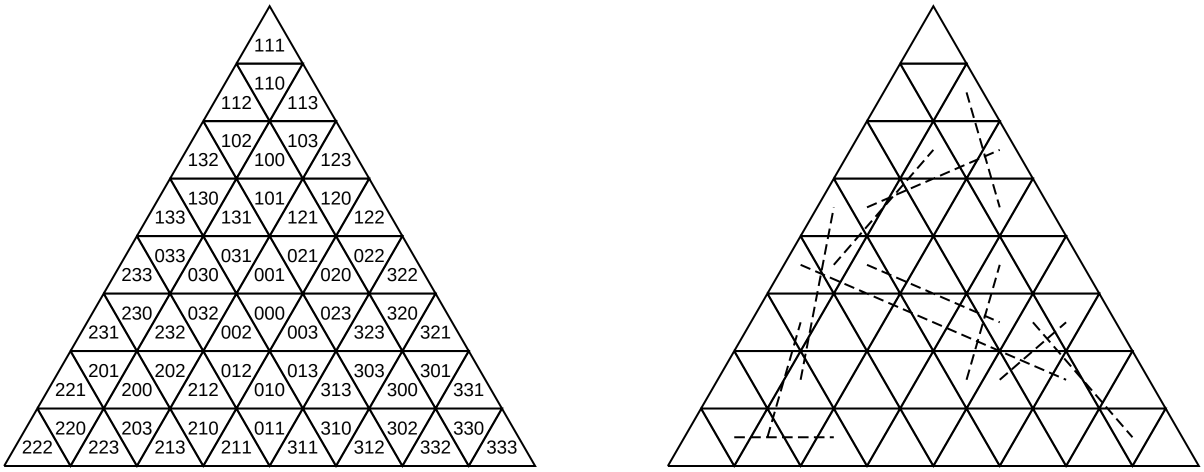

Figure 8: Dutton ordering: discontinuities up to Level 3

digits to its label. The relationship between ancestors and descendants in Otoo and Zhu's scheme is no longer so direct. For example, the children of 02 are 012, 031, 032, and 022. These problems arise from the fact that the subdivision is based on the triangle, while the labeling is based on the quadtree.

Otoo and Zhu's ordering is discontinuous, but less so than the other schemes discussed. In addition, since the ordering starts at one vertex of a triangle, and ends at another vertex, a circuit of the sphere can be created by properly linking triangles of the initial spherical partition (in such a way that the path *exits* one triangle at the same vertex where it *enters* the next triangle in the path). Recall that orderings based on the other three labelings start (or end) at the center of a subdivided triangle.

3.5 Discontinuities of Dutton-type orderings

The labeling schemes of Dutton, Goodchild and Shiren, and Fekete and Davis are close variations. They lead to very similar orderings, with the same number and total length of discontinuities. We will examine Dutton's, as representative of the three.

Dutton's scheme has 0 discontinuities between 4 triangles at Level 1, and 2 discontinuities between 16 triangles at Level 2 (Figure 3). At Level 3 it has 11 discontinuities between 64 triangles (Figure 8).

Proposition 1 *The number of discontinuities in Dutton's ordering at Level* n *is given by*

$$d_{dut}(n) = 3 \cdot 4^{n-2} - 1, \ \textit{for } n \geq 2.$$

All proofs are found in the Appendix.

We can get a sense of how discontinuous Dutton's ordering is by comparing the number of discontinuities to the total number of sequential cell pairs at Level n, given by $4^n - 1$. Taking the limit of the expression in Proposition 1 yields the *discontinuity rate.*

Proposition 2 *The discontinuity rate of Dutton's ordering is given by*

$$\lim_{n\to\infty} \frac{d_{dut}(n)}{4^n - 1} = \frac{3}{16},$$

where $d_{dut}(n)$ is the number of discontinuities in Dutton's ordering at Level n.

In practice, convergence is rapid, so this is a good approximation even when n is relatively small.

We can also compute the length of the jumps. We assume that the side of the bounding equilateral triangle is of unit length. In addition, recalling that Dutton-type orderings begin from the center of a triangle and end at a vertex, we will measure distances starting from highest vertex of the triangle at the start of the discontinuity, and ending at the center of the triangle at the end of the discontinuity.

Proposition 3 *The total length of the discontinuities in Dutton's ordering at Level n is given by*

$$l_{dut}(n) = \frac{7 \cdot 2^{n-3} - 2}{\sqrt{3}}, \text{ for } n \geq 2.$$

Proposition 4 *The limiting proportion of the total length of discontinuities to the total length of the curve corresponding to Dutton's ordering is given by*

$$\lim_{n\to\infty} \frac{l_{dut}(n)}{l_{dut}(n) + lc_{dut}(n)} = \frac{7\sqrt{3}}{11\sqrt{3} + 12} \approx 0.39,$$

where $lc_{dut}(n)$ signifies the total length of the continuous portions of Dutton's ordering at Level n.

Dutton's ordering of the sphere starts with the eight base triangles of an octahedron. In addition to the discontinuities *within* each base triangle, there is a discontinuity *between* each of the base triangles. This adds seven large discontinuities to the ordering. (This also applies to Goodchild and Shiren's ordering.) Fekete and Davis's ordering, based on the twenty-sided icosahedron, adds nineteen such discontinuities, but each is much smaller.

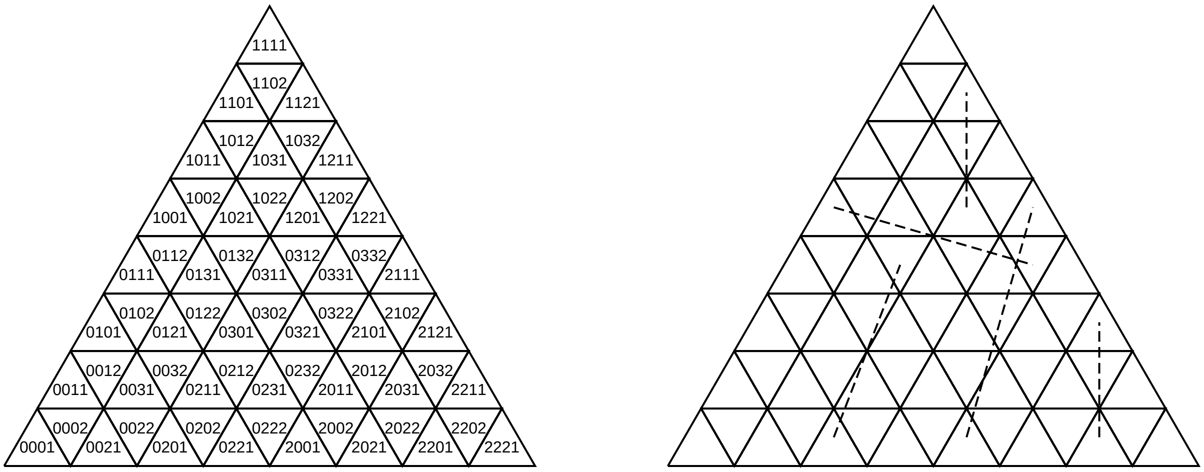

Figure 9: Otoo and Zhu ordering—discontinuities up to Level 3

3.6 Discontinuities of Otoo and Zhu's semi-quadcode ordering

Otoo and Zhu's semi-quadcode (SQC) ordering leads to fewer discontinuities than Dutton's ordering. Figure 9 shows discontinuities up to Level 3. Level 1 has 0 discontinuities among 4 triangles. Level 2 has 1 discontinuity among 16 triangles. Level 3 has 5 discontinuities among 64 triangles. Level 4 (not shown) has 19 discontinuities among 256 triangles.

Recall that Otoo and Zhu's SQC labeling scheme is based on a quadtree labeling. We will use the term *quadcode* (QC) to refer to the quadtree labeling.

Proposition 5 *The number of discontinuities in Otoo and Zhu's ordering at Level n is*

$$d_{sqc}(n) = 4^{n-2} + 2^{n-2} - 1, \ \textit{for } n \geq 2.$$

Again, we can get a sense of how discontinuous the SQC ordering is by comparing the number of discontinuities at some Level n to the number of sequential cell pairs at that level, given by $4^n - 1$. Taking the limit from the expression in Proposition 5 yields the following proposition:

Proposition 6 *The discontinuity rate of Otoo and Zhu's semi-quadcode ordering is given by*

$$\lim_{n \to \infty} \frac{d_{sqc}(n)}{4^n - 1} = \frac{1}{16},$$

where $d_{sqc}(n)$ is the number of discontinuities in Otoo and Zhu's semi-quadcode ordering at Level n.

Therefore, Otoo and Zhu orderings have, in the limit, one-third as many discontinuities as Dutton-type orderings.

We can also compute the length of the jumps in the ordering. We assume that the side of the bounding equilateral triangle is of unit length. Recalling that Otoo and Zhu orderings begin and end at triangle vertices, we will measure distances starting from the highest vertex of the triangle at the start of the discontinuity, and ending at the lowest vertex of the triangle at the end of the discontinuity.

Proposition 7 *The total length of the discontinuities in Otoo and Zhu's ordering at Level n is given by*

$$l_{sqc}(n) = 3 \cdot 2^{n-3} - 1, \text{ for } n \geq 3.$$

In addition to having fewer discontinuities than Dutton-type orderings, Otoo and Zhu orderings have a smaller total discontinuity length. In the limit, we have

$$\frac{l_{\text{dut}}}{l_{\text{sqc}}} \to \frac{7\sqrt{3}}{9} \approx 1.35.$$

Proposition 8 *The limiting proportion of the total length of discontinuities to the total length of the curve corresponding to Otoo and Zhu's ordering is given by*

$$\lim_{n\to\infty} \frac{l_{sqc}(n)}{l_{sqc}(n) + lc_{sqc}(n)} = \frac{9}{10\sqrt{3}+9} \approx 0.34,$$

where $lc_{sqc}(n)$ signifies the total length of the continuous portions of Otoo and Zhu's ordering at Level n.

Otoo and Zhu's ordering is the only one of the four orderings we have discussed in which the lowest and highest points within a triangle are at distinct vertices. Therefore, it is possible to link the eight base triangles of their initial octahedral partition of the sphere in such a way that a circuit is created with no additional discontinuities.

4 Continuous Indices of the Globe

4.1 Spacefilling curve on a sphere

We have developed continuous ordering schemes, based on spacefilling curves, for the hierarchical subdivision of a spherical surface. A *spherical spacefilling curve* is a continuous mapping from a one-dimensional interval, to the points on the spherical surface. Continuous orderings based on spacefilling

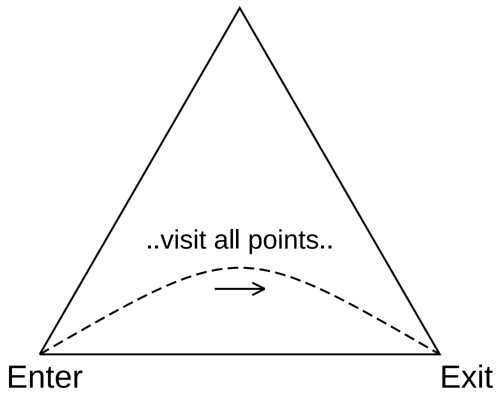

Figure 10: How a spacefilling curve orders each triangle

curves have proven useful in heuristics related to a number of spatial, combinatorial, and logistical problems (Bartholdi and Platzman 1982, 1988, Bartholdi, Platzman, Collins and Warden 1983, Nulty 1993, Platzman and Bartholdi 1989, Sagan 1994, p. 30).

The global data models we have so far discussed have discontinuous spatial orderings, and preserve nearness in tendency only: cells with similar labels *tend* to be near each other. In a continuous spatial ordering, cells with sequential labels are spatially adjacent, points with close index values are near on the sphere, and spatially adjacent cells tend to have similar labels. Moreover, for an equal-area subdivision of the sphere, each cell at the same level of a subdivision has the same area, and equal one-dimensional intervals correspond to equal size, contiguous areas on the sphere.

We are interested in spacefilling curves that begin at one vertex, and end at a different vertex. We will refer to these vertices as *entry* and *exit* vertices, respectively. The spacefilling curve is said (loosely speaking) to visit all the points within the triangle, starting from the entry vertex, and ending at the exit vertex (Figure 10).

To create a continuous path on a sphere, we partition the surface of the sphere into (spherical) triangular regions and order the space within each region according to a spacefilling curve. We sequence and orient the regions in such a way that the exit vertex of one triangle matches the entry vertex of the next triangle in the sequence. In this way, we create a continuous path through the surface. If, in addition, the first and last points coincide, it is a continuous circuit.

In common with the discontinuous orderings, spherical versions of the Platonic solids are reasonable choices for initial partitions in the continuous orderings. Figure 11 shows a spacefilling curve tracing a continuous circuit over an octahedral surface. The curve shown is one of a number of possible curves

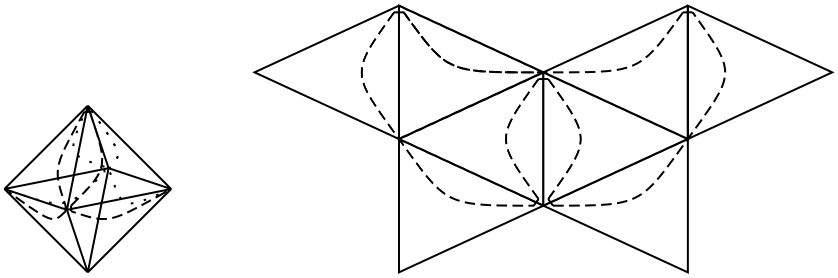

Figure 11: Spacefilling curve on an octahedral surface

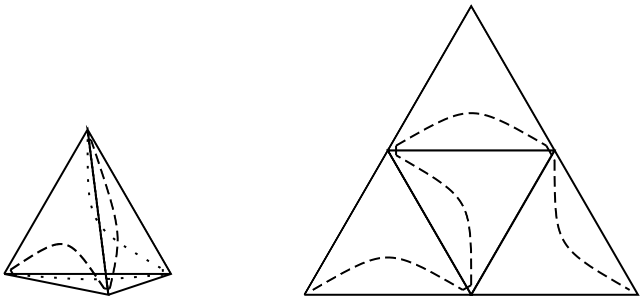

Figure 12: Spacefilling curve on a tetrahedral surface

that could have been drawn. Figure 12 shows a spacefilling curve tracing a continuous circuit over a tetrahedral surface. For flattened views of all the Platonic solids, see Laurini and Thompson (1992).

This is in the spirit of the earlier work on hierarchical global data models. It is more general in the sense that we do not specify a particular initial partition of the sphere (or equivalently, a base polyhedron), just that it be a triangular tessellation, with no vertices appearing in the middle of edges. We impose only very loose conditions on the way the subdivision itself is performed. Also, we use particular spacefilling curves in our examples, but other possibilities exist.

Cells within the hierarchical subdivision are labeled with codes indicating their relative position on a spacefilling curve. Because we are using a hierarchical data structure, these labels can identify arbitrarily small regions by matching points on the spacefilling curve with *sufficiently small* enclosing cells. For a map of cities on the earth, for example, we might be satisfied with a one-square mile enclosing cell around points representing cities. For a map of property lines, we would require much finer resolution. Converting between coordinates and spacefilling curve position is thus not completely reversible. We may not recover exactly the same point, but it will be *sufficiently close.* Conversions can be performed

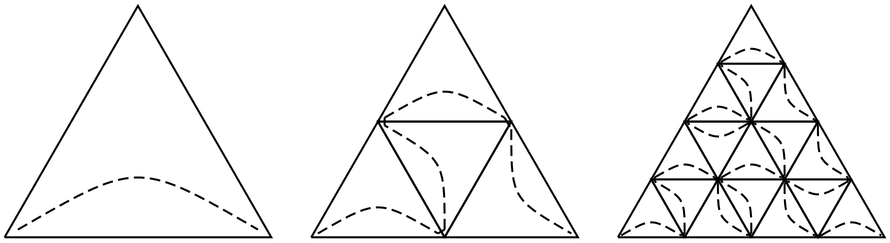

Figure 13: Quaternary spacefilling curve

in such a way that error does not accumulate. Of course, points expressed as finite-precision coordinates also have this sort of uncertainty; the true location of the entity referred to is within some region around the given point.

We present two examples in some detail. Both assume an initial partition of the spherical surface into triangular regions. The first example is based on a *quaternary subdivision*, in which each triangle is hierarchically subdivided into four subtriangles. The second is based on a *binary subdivision*, in which each triangle is hierarchically subdivided into two triangles.

4.2 Quaternary spacefilling curve

The *quaternary subdivision* refers to the hierarchical subdivision of a triangle into four subtriangles, as in Figure 1. Indexing schemes so far proposed for global data structures based on this subdivision have been discontinuous. We present a continuous scheme based on a spacefilling curve.

Figure 13 shows iterations of a spacefilling curve based on the quaternary subdivision. We will refer to this curve as the *Quaternary spacefilling curve* (QSFC). The curve is drawn to explicitly show the entry and exit points of the curve within each triangle. Figure 14 shows an alternate way to draw the same curve, emphasizing instead the order in which the triangles are traversed. Assuming the globe is divided into triangular faces, we order the space within each triangle according to the QSFC, oriented appropriately, to create a spherical spacefilling curve.

Below we describe procedures to generate cell labels, to convert between spatial coordinates and position on the QSFC, and to find cell neighbors. These procedures are based on an approach to creating algorithms for manipulating spacefilling curves that we discuss in greater detail in Bartholdi and Goldsman (n.d.b).

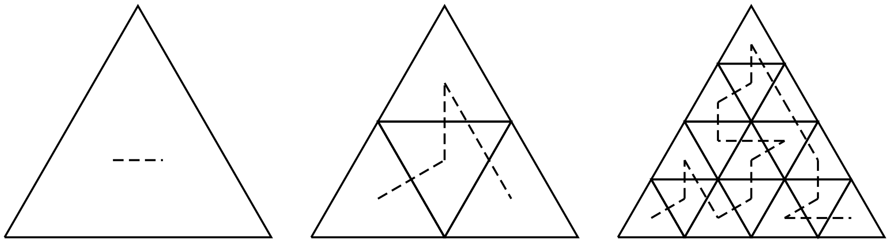

Figure 14: Quaternary spacefilling curve, alternate drawing

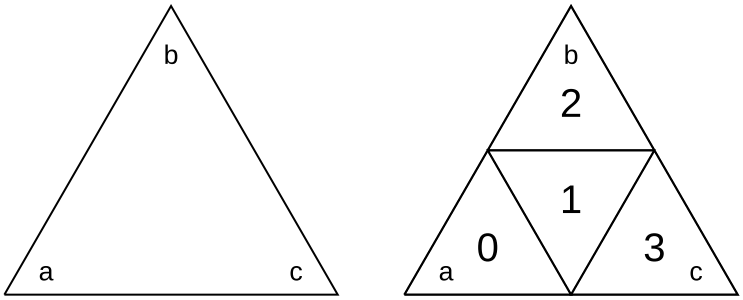

Figure 15: Labeling cells in the first level of subdivision

4.2.1 Ordering space according to the Quaternary curve

This procedure generates QSFC-based cell labels. We begin with a triangular cell with vertices labeled a, b, and c (Figure 15). The vertices imply a certain orientation of the cell. In particular, vertex a will be the lowest point, according to the ordering, and c will be the highest. We subdivide the cell into four subtriangles by choosing a point on each side and drawing line segments connecting each pair of points. (Typical choices are midpoints of each side, so that subcells are equivalent on the plane, and approximately equivalent on the sphere.) The label of each subcell is constructed as follows:

1. The cell containing vertex a: initialize its label to 0.
2. The cell containing vertex b: initialize its label to 2.
3. The cell containing vertex c: initialize its label to 3.
4. The central cell: initialize its label to 1.

Next label the nine unlabeled vertices of the four triangles as follows (see Figure 16):

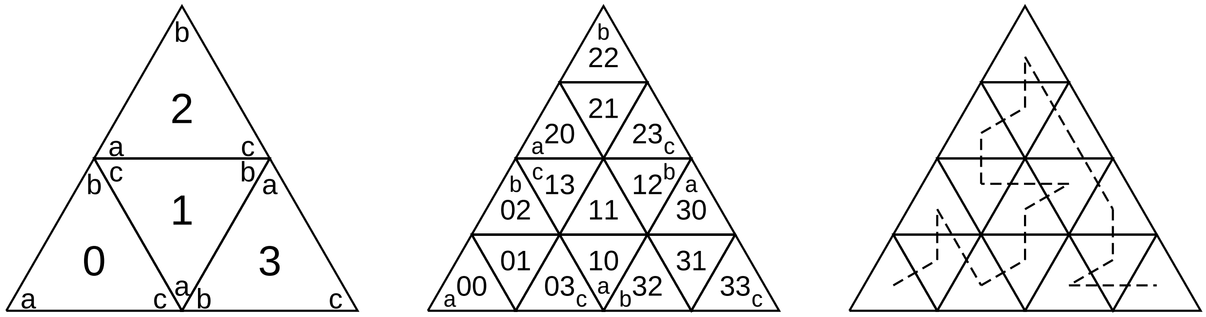

Figure 16: Quaternary spacefilling curve cells at Level 2

1. Vertices between a and b are labeled (in order): bca
2. Vertices between b and c are labeled (in order): cba
3. Vertices between c and a are labeled (in order): bac

Continue by subdividing each triangle in the same way, adding a digit to the right of the label of each cell, at each new level in the subdivision. Figure 16 shows the second level in the subdivision, with its associated spatial ordering. At this point each cell is identified by a two-digit label. This process continues until "sufficient precision" is reached.

The labels define an ordering of the space within the entire triangular cell. The ordering is continuous since cells with adjacent labels are adjacent in space; they share an edge or a vertex. The converse also tends to be true: cells near in space tend to have close labels. In the limit, every point in the space is ordered in this way. Figure 17 shows the labeling of Level 4 of the QSFC.

4.2.2 Properties of Quaternary curve ordering

A labeling scheme based on the QSFC establishes a continuous ordering of the space within a triangular region, and leads to a continuous circuit over the surface of a sphere. QSFC ordering is not stable under extensions to its domain, but this property is not important for global data structures. The index has the more important property of order-consistency across levels of the subdivision: regions are ordered the same, relative to each other, at different levels of the subdivision.

In addition, ancestors and descendants of regions can be found directly, and it is relatively simple to convert between QSFC labels and spatial coordinates (Section 4.2.4), and to find neighbors (Section 4.2.5).

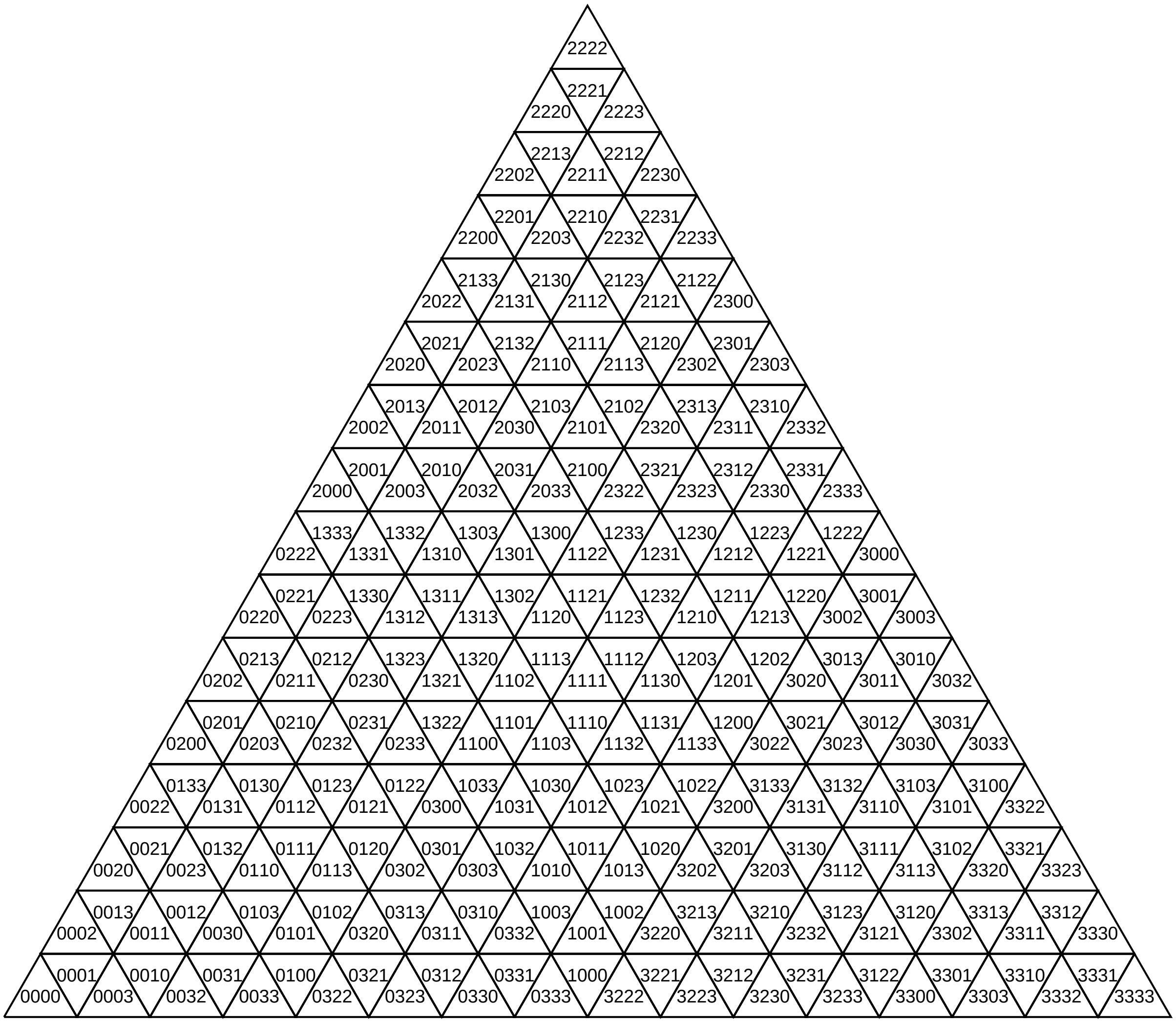

Figure 17: Level 4 ordering of the Quaternary spacefilling curve

4.2.3 Converting between Quaternary curve labels and coordinates

We determine the QSFC label corresponding to the spatial coordinates of a point, to desired precision, by finding a *sufficiently small* enclosing cell around the point. In finding the QSFC label of a point, we do not need to label all the cells in the space, but just the ancestor cells enclosing the point, so the amount of work is proportional to the precision of the result (in other words, the number of digits in the resulting label). When a point lies on the border between two cells at some level of the subdivision, we must choose one of the cells according to some rule, such as "always choose the smallest label."

Figure 18 shows the steps in finding the label of point P. In this example, we require precision to four quaternary digits, so we subdivide the triangle to four levels. The initial step subdivides the triangular cell into four subcells labeled 0, 1, 2, and 3. Cell 0 contains P, therefore the second step subdivides cell 0 into subcells 00, 01, 02, and 03. The third step subdivides cell 01 into 010, 011, 012, and 013. The final step subdivides cell 013 into 0130, 0131, 0132, and 0133. From the figure, the label of P is 0131.

4.2.4 Conversion procedure

Given a point P, for which we wish to find the corresponding QSFC label within a triangular region t:

1. Label t so that the entry vertex is a, the exit vertex is c, and the remaining vertex is b.

2. Subdivide t, as in Figure 15.

3. Find the subtriangle of t containing P. If sufficient precision has been reached then end. Otherwise, let this subtriangle now be called t; label as in the left-hand diagram of Figure 16; and return to 2.

4. End: The label of P is the label of the smallest enclosing cell.

The reverse procedure, to generate the coordinates of a point given its label, is similar to the procedure to generate the label. Vertex labeling proceeds as above. At each level of the subdivision, the left-most remaining digit of the label identifies the enclosing triangle at the current level of subdivision (that is, the triangle containing the point). This digit is deleted, and we proceed until no more digits remain. Again, the result is not a point, but a triangular cell of *sufficient precision* enclosing the point. We are indifferent to exactly which point in the cell is *the* point. If needed, some rule may be used to pick a particular point, such as "choose the average of the vertices of the enclosing cell."

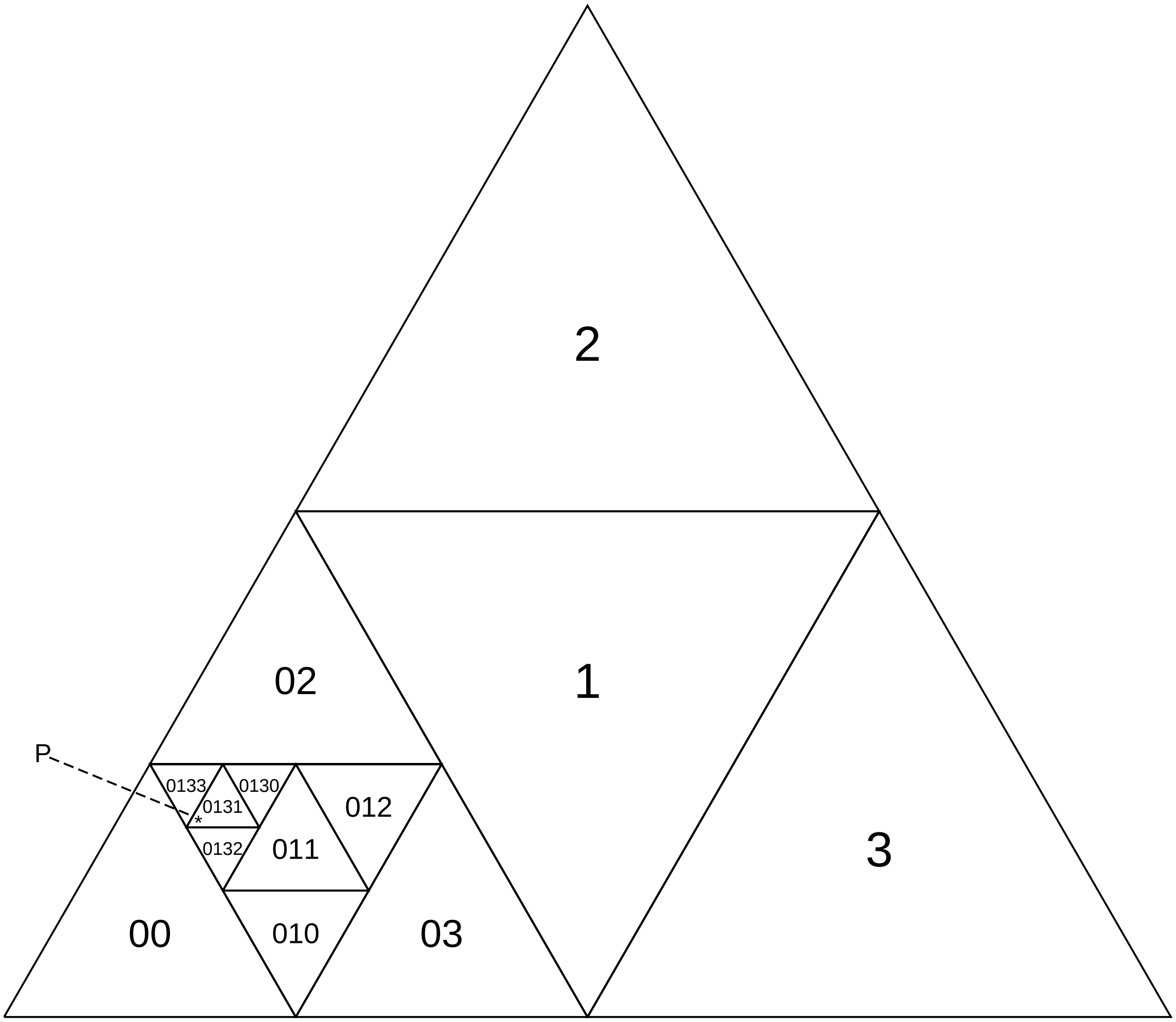

Figure 18: The label of point P after four subdivisions is 0131

4.2.5 Neighbor-finding

A useful operation is finding edge-adjacent neighbors of triangular cells. Most cells have three neighbors; cells on edges of the bounding region have either one or two neighbors. For example, in Figure 17, Cell 2221 is interior and has three neighbors: 2220, 2222, and 2223; Cell 2222 is on a boundary and has just one neighbor: 2221.

Suppose we wish to find all edge-adjacent neighbors of a particular cell A. We start by choosing a point on the interior of each of the edges of A. For each of these points, we search through the subdivision, as in the procedure of Section 4.2.4. Unless the point is on the bounding triangle, we will eventually find that the point lies on the boundary between two cells: One of the two cells will be A, and the other will be one of its neighbors. We continue by following the sequence of subdivisions to find the neighbor. Since no point on the interior of an edge can belong to more than two cells, this will happen at most one time for each of the three points. Since the work to traverse each path is proportional to the depth of the hierarchy, the work to find the neighbors is also proportional to the depth of the hierarchy. A somewhat more efficient procedure is to search for all three cells simultaneously, searching for cells that contain any of the points. The work is still proportional to the depth of the subdivision.

A similar procedure can be created to find all vertex-adjacent neighbors of A. In this case, we search for cells containing any of the vertices of A.

In many cases, there is a direct way to generate the labels of edge-adjacent neighbors, by observing certain label patterns. For example, a triangle whose label ends in the digit '1' is a central triangle within a group of 4. Therefore, its three neighbors can be found by exchanging the '1' with the digits '0', '2', and '3', respectively. Many such correspondences exist.

4.3 Sierpinski spacefilling curve

We present another continuous indexing scheme of the sphere based on the recursive *binary* subdivision of triangular regions. We hierarchically subdivide each triangle into two subtriangles, and then order triangles according to the Sierpinski spacefilling curve, to create a continuous ordering. Again, we imagine the Sierpinski spacefilling curve entering a triangle at an *entry* vertex, visiting every point in the triangle, and then exiting at a distinct *exit* vertex. Figure 19 shows iterations of the Sierpinski spacefilling curve. The curve is drawn to emphasize how it *enters* and *exits* each cell.

We describe a procedure to generate cell labels, and touch on the related functions of converting between spatial coordinates and position on the Sierpinski curve, and finding neighbors of cells.

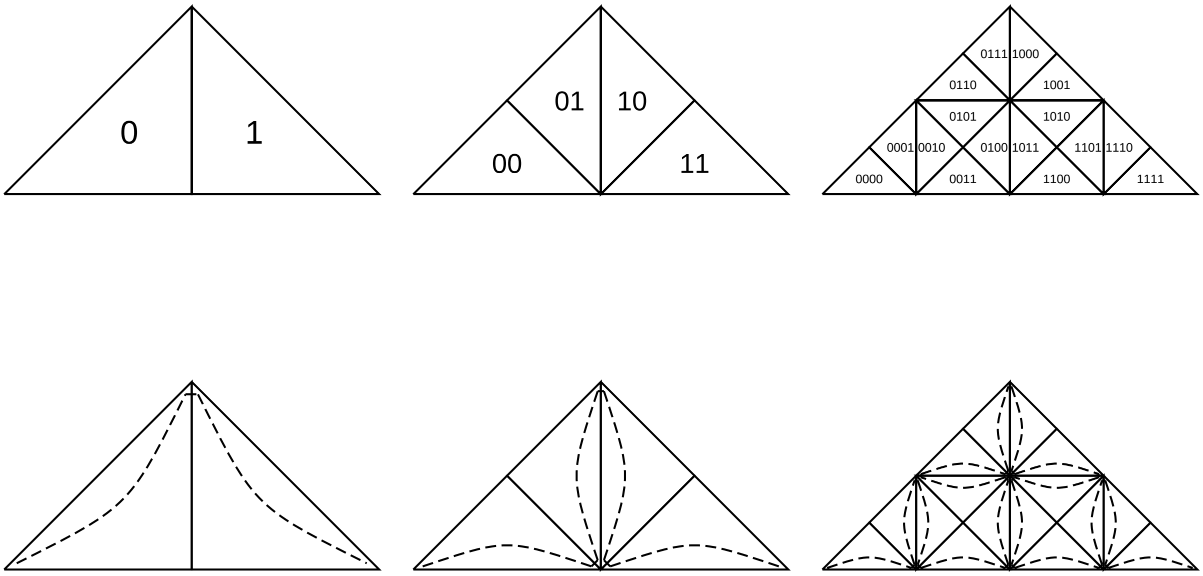

Figure 19: Sierpinski curve: Levels 1, 2, and 4

4.3.1 Ordering space according to the Sierpinski curve

We begin with a triangular cell with vertices labeled a, b, and c (Figure 20, left). The vertices imply a certain orientation of the cell. In particular, vertex a will be the lowest point, according to the ordering, and c will be the highest. To subdivide the cell into two subtriangles, we add a line segment from vertex b to some point on the side across from b. We typically choose the midpoint of the side, so that the subcells will be approximately equal, but this is not necessary. The label of the subtriangle containing vertex a is set to 0. The label of the subtriangle containing c is set to 1.

The shape of the curve (Figure 19) guides us in relabeling vertices. Within each cell, the entry

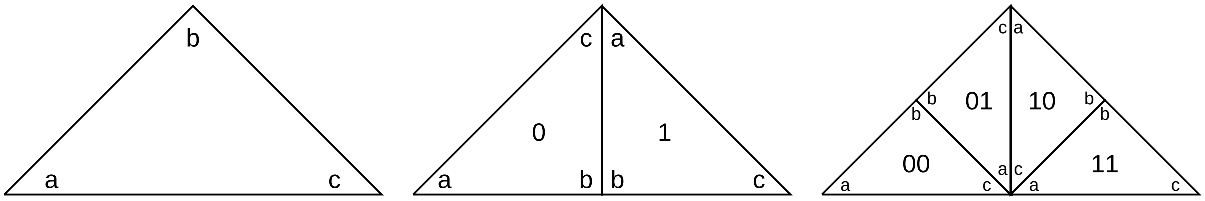

Figure 20: Labeling cells according to the Sierpinski curve

vertex is a, the exit vertex is c, and the intermediate vertex is b. This leads to the following mechanical relabeling rules: Vertex b is relabeled a in the triangle containing c, and c in the triangle containing a. The two remaining unlabeled vertices (on line segment ac of the parent cell) are both labeled b, as in the figure.

Each iteration proceeds in the same way. When a triangle is subdivided, the label of each of the subtriangles is created by adding a digit to the right of the label of the parent triangle. If the subtriangle contains vertex a, the digit 0 is added to the right of the label. If the subtriangle contains vertex b, the digit 1 is added to the right of the label. This process continues until the cells are sufficiently small, according to the application. The length of the cell label equals its level in the subdivision.

When all the cells on one level are labeled, one can trace the path of the underlying spacefilling curve. At any level of the subdivision, the triangular cells are ordered continuously, and each cell shares an edge with its predecessor and successor cells in the sequence. Figure 21 shows Level 5 of the Sierpinski spacefilling curve. The curve is drawn to emphasize how the cells are ordered.

Converting between labels and coordinates, and neighbor-finding proceed as in the case of the Quaternary curve (Sections 4.2.4 and 4.2.5). These functions are simple variations on the cell labeling procedure, with the amount of work required proportional to subdivision level.

4.3.2 Properties of the Sierpinski curve ordering

A labeling scheme based on the Sierpinski spacefilling curve establishes a continuous ordering of the space within a triangular region, and leads to the implementation of a spherical spacefilling curve. The Sierpinski curve in particular has been studied in connection with a number of spatial and combinatorial problems (see Section 4.1).

The Sierpinski ordering has certain other advantages. It is easier to convert between labels and spatial coordinates than any of the other orderings presented. It is also the most symmetric ordering. Like the Quaternary curve, the Sierpinski curve is order-consistent across subdivision levels. This leads to direct ancestor- and descendant-finding. Unlike the Quaternary curve, the Sierpinski label is stable in domain extension, but this property is not of much importance on the globe. The Sierpinski curve also has an advantage in neighbor-finding: cells with consecutive labels are edge-adjacent. This means that two edge-adjacent neighbors of all but the first and last cells can be found directly. The third neighbor of cells can be found as in Section 4.2.5.

An additional advantage of the Sierpinski curve is that, over a sphere, the binary subdivision starting from a cube results in equivalent subtriangles for the first 3 levels of the subdivision: The first subdivision

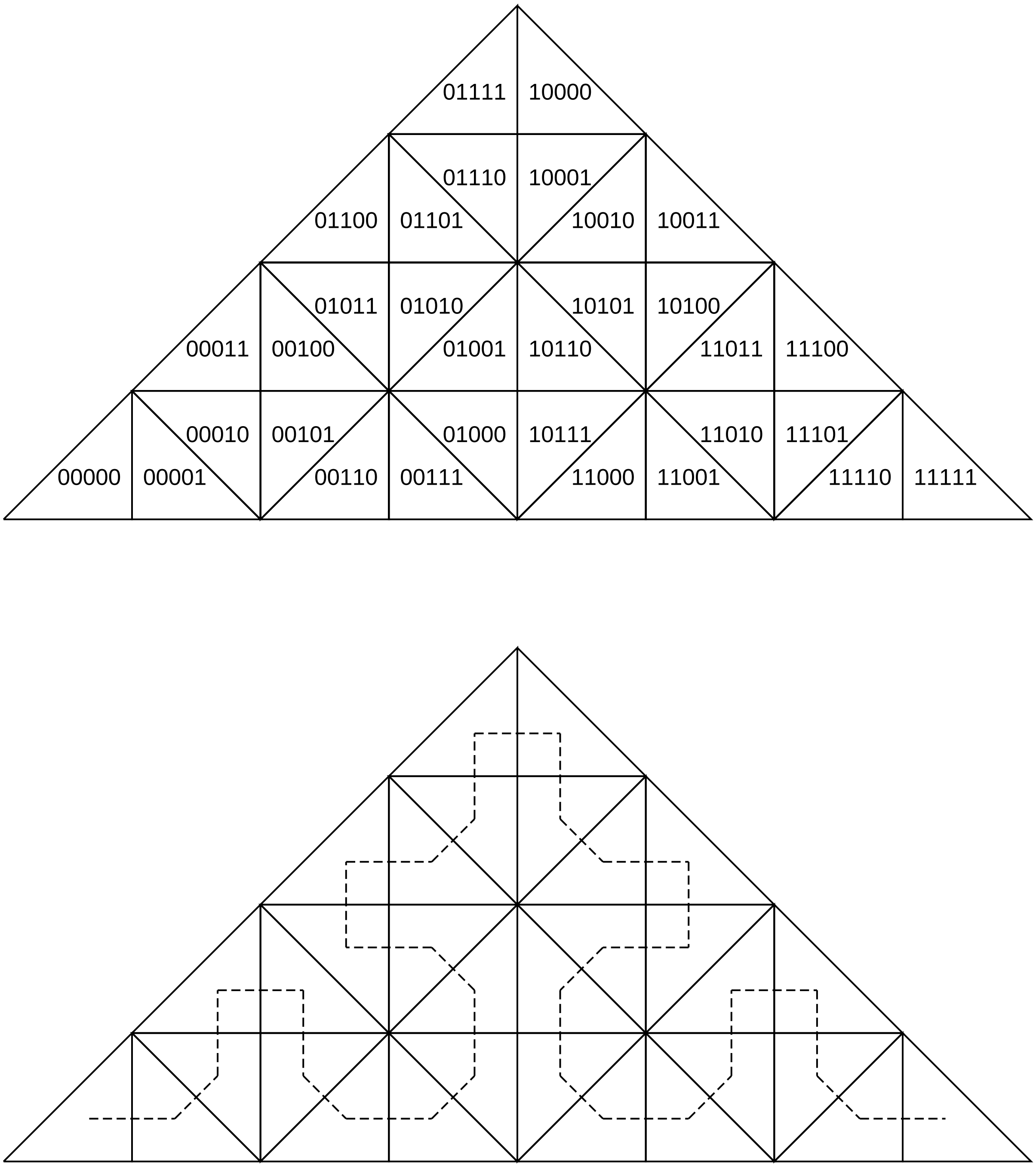

Figure 21: Level 5 ordering of the Sierpinski spacefilling curve

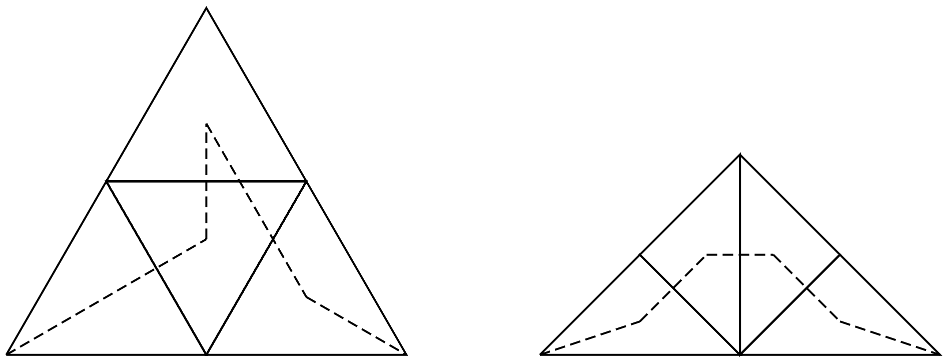

Figure 22: Level 1 Quaternary and Sierpinski orderings

divides the 6 spherical squares into 12 spherical isosceles triangles that are equivalent in size and shape. The second subdivision results in 24 equivalent spherical isosceles triangles. The third subdivision results in 48 equivalent spherical scalene triangles. Therefore, this leads to a data model based on 48 equivalent regions of the sphere. By contrast, a quaternary subdivision based on an octahedral partition of the sphere is immediately distorting to its initial eight spherical equilateral triangles. The binary subdivision thus makes it easier to derive the benefits of a fine initial partition, while starting from a simple initial partition.

5 Comparison of Continuous Versus Discontinuous Orderings

5.1 Lengths of curves based on the orderings

We have already derived expressions for the total length of the curves implied by the discontinuous orderings (as a function of level number) (Sections 3.5 and 3.6). We also derive similar expressions for the orderings implied by the Sierpinski and Quaternary spacefilling curves. For consistency with the discontinuous orderings, we measure curve length between central points of cells, starting from the lowest vertex of a cell and ending at the highest vertex (Figure 22). Also, all the orderings, aside from that based on the Sierpinski curve, are based on the subdivision of one triangle into four subtriangles. The Sierpinski ordering is based on the subdivision of one triangle into two subtriangles. In order to arrive at a similar basis for comparison, we will, in this section and Section 5.2 only, count levels in the Sierpinski ordering in such a way that two subdivisions count as a single level. Therefore, at each level, all the orderings quadruple their number of triangular cells.

Finally, since we demonstrate the Sierpinski curve on an isosceles right triangle with two sides of unit length, and the other orderings on equilateral triangles with sides of unit length, we calculate a multiplicative factor for the Sierpinski curve, so that all comparisons are on triangular regions of equal area.

Proposition 9 *The total length of the Sierpinski curve at Level n is given by*

$$l_{ssfc}(n) = 2^{n-1} \frac{s}{6} \left(4 + \sqrt{2} + 2\sqrt{5}\right),$$

where $s = 3^{\frac{1}{4}}/\sqrt{2}$ is the side length to equalize the area of the isosceles right triangle with an equilateral triangle of unit side length.

Proposition 10 *The total length of the Quaternary curve at Level n is given by*

$$l_{qsfc}(n) = 2^{n-1} \left(\frac{2\sqrt{3}}{3} + \frac{1}{2}\right),$$

We have derived expressions for the total length of a number of curves based on discontinuous and continuous orderings. For the purpose of comparison, we define normalized length N as

$$N = \lim_{n \to \infty} \text{(length of curve)}/2^n.$$

Proposition 11

$$\begin{aligned} N_{\textit{Dutton-type}} &\approx 1.29 \\ N_{\textit{Semi-quadcode}} &\approx 1.10 \\ N_{\textit{Sierpinski}} &\approx 0.77 \\ N_{\textit{Quaternary}} &\approx 0.83. \end{aligned}$$

The continuous orderings lead to shorter curves. The Sierpinski spacefilling curve performs best, and the Quaternary spacefilling curve next best.

5.2 Average Data File Storage Distance metric

Given an ordering of the cells of a subdivision, the *Average Data File Storage Distance* (ADFSD) is defined as the average difference between the rankings of cells that share an edge (Goodchild and Shiren 1992). It is therefore a measure of adjacency preservation. A low value for this measure indicates that neighboring cells tend to have similar rankings; in other words, adjacency tends to be preserved. A

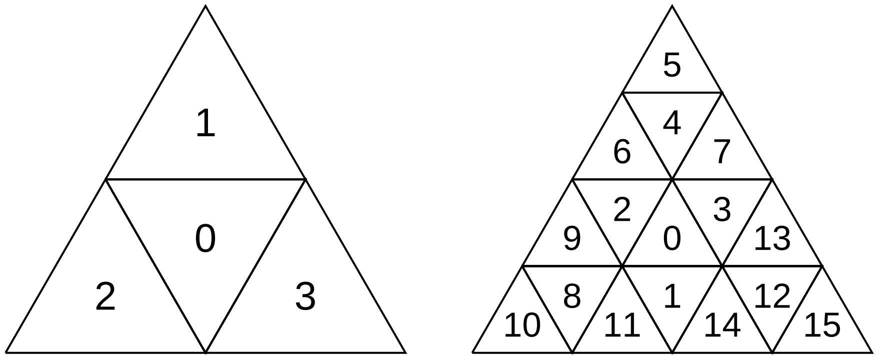

Figure 23: Goodchild and Shiren's Level 1 and 2 orderings

high value indicates that neighbors tend to have very different rankings. Goodchild and Shiren call the ADFSD "one of the important indices in data file structures for large geographical information systems," and cite its application to problems involving data compression and the prediction of access time.

The ADFSD is calculated for a given ordering, at a given subdivision level, by taking the sum of the absolute differences in rank of all pairs of neighboring cells, and dividing by the total number of such pairs.

For example, Goodchild and Shiren's Level 1 ordering has three pairs of neighboring triangles, with ranks of: $(0,1)$, $(0,2)$, and $(0,3)$. The sum of the absolute differences in rank is 6. Therefore, the ADFSD at this level is equal to 2. Goodchild and Shiren's Level 2 ordering has 18 pairs of neighboring triangles, with total sum of differences equal to 72. Therefore, the ADFSD at this level is equal to 4 (Figure 23). In general, the ADFSD of Goodchild and Shiren's ordering at Level j is 2^j (Goodchild and Shiren 1992). In fact, all of the *Dutton-type* orderings we have discussed (those of Dutton, Goodchild and Shiren, and Fekete and Davis) have the same ADFSD.

For completeness, we also derive the ADFSD for the semi-quadcode (SQC) ordering of Otoo and Zhu. This was the one *non-order-consistent* discontinuous ordering that we examined – that is, this ordering has the undesirable property that the order of regions or points may change at different levels of the subdivision.

Proposition 12 *For the semi-quadcode ordering*

$$ADFSD_{sqc} = \frac{34 \cdot 8^n - 21 \cdot 4^n - 7 \cdot 2^n - 6}{63 \cdot 2^n (2^n - 1)}.$$

We compare the ADFSD of our continuous orderings with the ADFSD of the discontinuous orderings. Our Quaternary spacefilling curve ordering and the discontinuous orderings are all based on the same quaternary subdivision of one triangle into four subtriangles, so it is straightforward to compare their ADFSD metrics directly. Our Sierpinski spacefilling curve ordering is based on the subdivision of one triangle into two subtriangles. As in Section 5.1, we count levels in the Sierpinski ordering in such a way that two subdivisions count as a single level, in order to arrive at a similar basis for comparison. This yields the same number of triangles and adjacent neighbors for all the orderings, at each level.

Proposition 13 *In the hierarchical subdivisions of the Dutton-type and the SQC orderings, and the Sierpinski, and Quaternary curves, the number of pairs of cell neighbors at Level n is given by*

$$p(n) = 3 \cdot 2^{n-1}(2^n - 1). \tag{1}$$

We derive the ADFSD for the Sierpinski and Quaternary spacefilling curves.

Proposition 14 *For the Sierpinski spacefilling curve ordering*

$$ADFSD_{ssfc} = \frac{2^n(2^n - 2^{-1})}{3(2^n - 1)}.$$

Proposition 15 *For the Quaternary spacefilling curve ordering,*

$$ADFSD_{qsfc} = \frac{7 \cdot 2^n(2^n - 6)}{12(2^n - 1)}.$$

The ordering based on the Sierpinski curve performs best and the Dutton-type orderings worst, according to this statistic; the ADFSD of Sierpinski approaches 1/3 that of the Dutton-type orderings, for large n. The SQC ordering does just slightly better than the Quaternary curve ordering (see Table 1, below).

5.3 Sequential-cell analysis

Two cells that have been ordered sequentially, according to some index, may be spatially adjacent or spatially non-adjacent. For each of the indices we have discussed, we measure how well an ordering preserves proximity information by determining the proportions of such cell pairs that are: edge adjacent, vertex adjacent, and non-adjacent (and so comprise a discontinuity in the index). We earlier defined the *discontinuity rate* of an ordering as the proportion of non-adjacent pairs as Level $n \to \infty$ (Section 3.5). We similarly define *edge-adjacency rate* and *vertex-adjacency rate* as the proportions of pairs that are edge-adjacent and vertex-adjacent, respectively.

Proposition 16 *For the Sierpinski spacefilling curve ordering: The edge-adjacency rate is* 1*; the vertex-adjacency rate is* 0.

Table I: Comparison of Orderings

	Sierpinski	Quaternary	Dutton-type	Otoo-Zhu
Continuous	yes	yes	no	no
Order-Consistent	yes	yes	yes	no
Normalized Curve Length	0.77...	0.83...	1.29...	1.10...
Discontinuous Proportion				
of Curve Length (in limit)	0	0	0.39...	0.34...
ADFSD/2^n (in limit)	1/3	7/12	1	34/63
Sequential-cell analysis				
· Discontinuity Rate	0	0	3/16	1/16
· Edge-adjacency Rate	1	2/3	1/4	3/4
· Vertex-adjacency rate	0	1/3	9/16	3/16

Proposition 17 *For the Quaternary spacefilling curve ordering: The edge-adjacency rate is* $2/3$*; the vertex-adjacency rate is* $1/3$.

Proposition 18 *For Dutton-type orderings at Level* $n \geq 2$*: The edge-adjacency rate is* $1/4$*; the vertex-adjacency rate is* $9/16$.

Proposition 19 *For Otoo and Zhu (SQC) orderings at Level* $n \geq 2$*: The edge-adjacency rate is* $3/4$*; the vertex-adjacency rate is* $3/16$.

The outcome for sequential-cell analysis is similar to that of the ADFSD metric: The Sierpinski curve performs best; Dutton-type orderings perform worst; the SQC ordering has slightly more edge-adjacencies than the Quaternary curve ordering, but is discontinuous.

6 Conclusions

Table 1 summarizes our results. The continuous orderings in general outperform the discontinuous orderings. The Sierpinski curve performs best by all of the measures. This curve produces the shortest orderings, has the lowest Average Data File Storage Distance, and all of its sequential cells are edge-adjacent.

Our approach produces indices with a number of other desirable characteristics, including order-consistency across levels of a hierarchical subdivision, and simple ancestor and descendant finding. We have also described simple and efficient methods for labeling subdivisions, converting between spatial coordinates and spacefilling curve position, and neighbor-finding. These characteristics are useful in implementing data structures and indices for storage, retrieval, and manipulation of geographic and other surface data.

We have assumed certain regular initial partitions of the sphere, and regular subdivisions, but many other possibilities exist. In principle, our approach is not tied to any particular initial partition, and could be supported by a wide variety of hierarchical subdivisions and spacefilling curves, in addition to those we have discussed.

In related work, we explore the creation of continuous indices (describing paths and circuits) on general, irregular planar triangulations, and find conditions for their existence (Bartholdi and Goldsman n.d.a), and the creation of general procedures for converting between spacefilling curve labels and coordinates, describing a family of procedures that are more flexible and straightforward than current procedures (Bartholdi and Goldsman n.d.b).

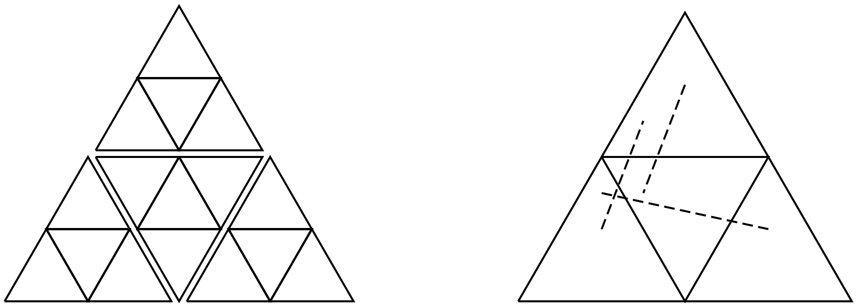

Figure 24: Dutton ordering discontinuities, deconstructed: A Level n ordering shown as the union of four Level $n-1$ orderings; three new discontinuities are added where the $n-1$ ordering link (when $n > 2$).

7 Appendix: Proofs

Proof of Proposition 1

We observe that Dutton's Level n ordering can be decomposed into four Level $n-1$ ordering. When $n > 2$, the total number of discontinuities at Level n is four times the number of discontinuities at Level $n-1$, plus three additional discontinuities where the Level $n-1$ orderings link. In other words, $d_{\mathrm{dut}}(n) = 4d_{\mathrm{dut}}(n-1) + 3$, for $n \geq 3$. This is shown in Figure 24; the dashed lines indicate the discontinuities between the Level $n-1$ orderings.

Using this recursive expression, and the fact that $d_{\mathrm{dut}}(2) = 2$, the proposition may be readily verified by standard induction arguments.

Proof of Proposition 3

The proof follows reasoning similar to that of Proposition 1. For any $n > 2$, we observe that the total length of discontinuities is four times the length of discontinuities at Level $n-1$, shrunk by $1/2$, plus three additional discontinuities where the Level $n-1$ orderings link. It is easy to verify that the total length of these three new discontinuities is $2\sqrt{3}/3$ (measuring from vertex to center point). In other words,

$$l_{\mathrm{dut}}(n) = \frac{4 \cdot l_{\mathrm{dut}}(n-1)}{2} + \frac{2\sqrt{3}}{3}.$$

When $n = 2$, Dutton's ordering has 2 discontinuities, and it is straightforward to show that $l_{\mathrm{dut}}(2) = \sqrt{3}/2$. The proposition readily follows.

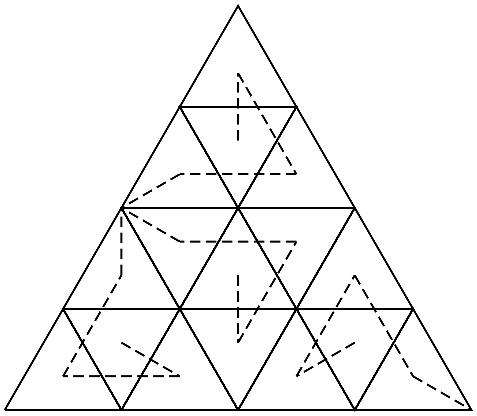

Figure 25: The continuous parts of Level 2 Dutton ordering

Proof of Proposition 4

We assume bounding triangle side length of 1. We measure curve lengths between center points, and we start at the lowest vertex of the bounding region and end at the highest. Discontinuities start at the highest vertex of a region and end at the lowest of the next region (Figure 25). It is simple to verify that $lc_{\text{dut}}(1) = 1 + \frac{\sqrt{3}}{3}$. Since each Level n is composed of four copies of the Level $n-1$, each shrunk in half, then it is straightforward to show that

$$lc_{\text{dut}}(n) = 2^{n-1}\left(1 + \frac{\sqrt{3}}{3}\right).$$

The result follows directly from $l_{\text{dut}}(n) = (7 \cdot 2^{n-3} - 2)/\sqrt{3}$ (Proposition 3).

Proof of Proposition 5

We observe (Figure 26) that at Level n, Otoo and Zhu's ordering can be decomposed into two Level $n-1$ Otoo and Zhu orderings (labeled *SQC*), and one Level $n-1$ quadtree ordering (labeled *QC*). The dashed lines in the figure indicate the discontinuities between the Level $n-1$ orderings.

For $n > 2$, the number of discontinuities in the Level n SQC equals the sum of the number of discontinuities in the three Level $n-1$ orderings, plus two additional discontinuities where the three $n-1$ orderings link, as in the figure. (Note that the number of discontinuities in a QC ordering is not changed when triangles are divided to create a SQC ordering, as in Figure 6.)

Letting $d_{\text{qc}}(n)$ signify the number of discontinuities in a QC ordering at Level n, we express this

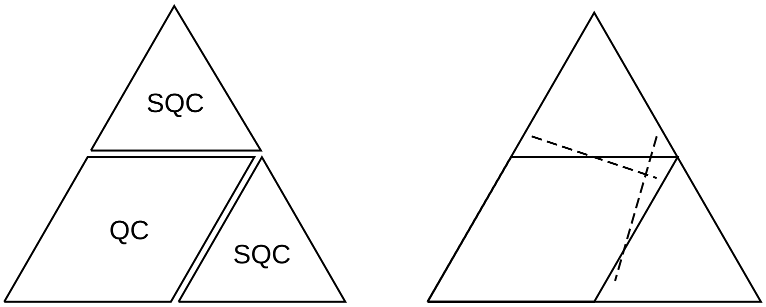

Figure 26: Semi-quadcode (SQC) ordering discontinuities, deconstructed: A Level n ordering shown as the union of two Level $n-1$ orderings and one $n-1$ quadtree; two new discontinuities are added where the orderings link (when $n > 2$).

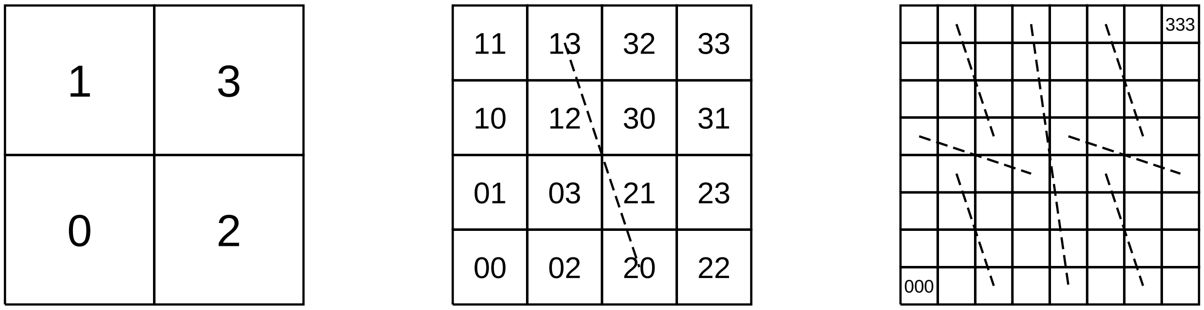

Figure 27: Quadcode (QC) discontinuities, Levels 1, 2, and 3

relationship as

$$d_{\text{sqc}}(n) = 2d_{\text{sqc}}(n-1) + d_{\text{qc}}(n-1) + 2, \text{ for } n \geq 3, \tag{2}$$

We seek an expression for d_{qc}. We note that $d_{\text{qc}}(1) = 0$, $d_{\text{qc}}(2) = 1$, and $d_{\text{qc}}(3) = 7$ (Figure 27).

For the purpose of counting discontinuities, a Level n QC ordering can be thought of as the combination of four Level $n-1$ QC orderings. Then the total number of discontinuities is the sum of the number of discontinuities within each of the four Level $n-1$ QCs, plus three additional discontinuities where the Level $n-1$ orderings are linked (Figure 28). In other words,

$$d_{\text{qc}}(n) = 4d_{\text{qc}}(n-1) + 3, \text{ for } n \geq 3.$$

Using this recursive expression, and the fact that $d_{\text{qc}}(2) = 2$, we can show by induction that

$$d_{\text{qc}}(n) = 2^{2n-3} - 1, \text{ for } n \geq 3. \tag{3}$$

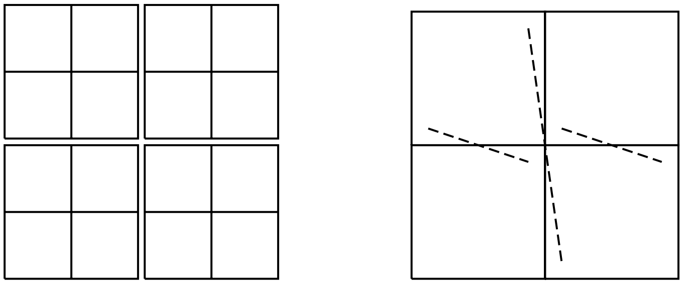

Figure 28: Quadcode (QC) ordering discontinuities, deconstructed: A Level n ordering shown as the union of four Level $n-1$ orderings; three new discontinuities are added where the orderings link (when $n > 2$).

Using expressions (2) and (3), and the fact that $d_{\mathrm{sqc}}(2) = 1$, then the proposition itself may be readily verified by induction arguments.

Proof of Proposition 7

The proof follows reasoning similar to the proof of Proposition 5. For any $n > 3$, we observe that the total length of discontinuities is two times the total length of discontinuities at Level $n-1$, shrunk by 1/2, plus the total length of quadcode discontinuities at Level $n-1$ (that is, $l_{\mathrm{qc}}(n-1)$) shrunk by 1/2, plus two additional discontinuities where the Level $n-1$ orderings link; it is easy to verify that the total length of these two new discontinuities is 1 (measuring from vertex to vertex). In other words,

$$l_{\mathrm{sqc}}(n) = \frac{2 \cdot l_{\mathrm{sqc}}(n-1)}{2} + \frac{l_{\mathrm{qc}}(n-1)}{2} + 1, \tag{4}$$

It is straightforward to show (using an inductive argument, for example) that $l_{\mathrm{qc}}(n) = 3 \cdot 2^{n-2} - 2$. Substituting this expression into (4), we have

$$l_{\mathrm{sqc}}(n) = l_{\mathrm{sqc}}(n-1) + 3 \cdot 2^{n-4}, \text{ for } n \geq 3. \tag{5}$$

From (5) and the fact that $l_{\mathrm{sqc}}(2) = 1/2$, the proposition is readily verified by induction.

Proof of Proposition 8

We assume bounding triangle side length of 1. We measure curve lengths between center points, and we start at the lowest vertex of the bounding region and end at the highest. Discontinuities start at the highest vertex of a region and end at the lowest of the next region (Figure 29).

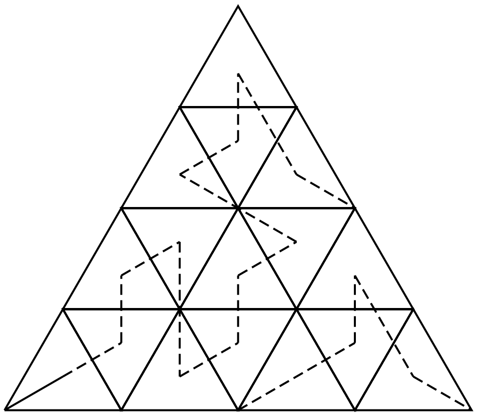

Figure 29: The continuous parts of Level 2 Otoo and Zhu ordering

Following similar reasoning to the proofs of Propositions 5 and 7, we start by noting that the total length of the continuous portions of the ordering at Level n is twice the length at Level $n-1$, shrunk by 1/2, plus the total length of the continuous portions of the quadcode ordering at Level $n-1$ (that is, $lc_{\mathrm{qc}}(n-1)$) shrunk by 1/2. In other words,

$$lc_{\mathrm{sqc}}(n) = \frac{2 \cdot lc_{\mathrm{sqc}}(n-1)}{2} + \frac{lc_{\mathrm{qc}}(n-1)}{2}, \tag{6}$$

It is simple to verify that

$$lc_{\mathrm{sqc}}(1) = \frac{2\sqrt{3}}{3} + \frac{1}{2}. \tag{7}$$

To find the length of the continuous parts of the quadcode curve, we start by determining that $lc_{\mathrm{qc}}(1) = \frac{5\sqrt{3}}{3}$. Since each Level n curve is composed of four copies of the Level $n-1$ curve, each shrunk in half, then it is straightforward to show that

$$lc_{\mathrm{qc}}(n) = 2^n \frac{5\sqrt{3}}{6}.$$

Substituting this expression and (7) into (6), we use standard induction arguments to show that

$$lc_{\mathrm{sqc}}(n) = \frac{5\sqrt{3}}{12} \cdot 2^n + \frac{1}{2} - \frac{\sqrt{3}}{6}.$$

The result follows directly from $l_{\mathrm{sqc}}(n) = 3 \cdot 2^{n-3} - 1$ (Proposition 7).

Proof of Proposition 9

It is simple to verify that

$$l_{\mathrm{ssfc}}(1) = \frac{s}{6}\left(4 + \sqrt{2} + 2\sqrt{5}\right),$$

where s is the length of the equal sides of the isosceles right triangle. We can also readily confirm that the area of the isosceles right triangle is equal to the area of an equilateral triangle with sides of unit length, when $s = 3^{\frac{1}{4}}/\sqrt{2}$.

When measuring curve length starting at one vertex and ending at another, as we have indicated, the length of the curve at each Level n is four times the length of the curve at Level $n-1$, divided by two. The proposition follows directly.

Proof of Proposition 10

It is simple to verify that

$$l_{\text{qsfc}}(1) = \frac{2\sqrt{3}}{3} + \frac{1}{2}.$$

Again, the proposition follows directly from the fact that the length of the curve at each Level n is four times the length of the curve at Level $n-1$, shrunk by half.

Proof of Proposition 12

This follows reasoning similar to the proofs of Propositions 5 and 7. That is, begin by dividing a Level n SQC ordering into two Level $n-1$ SQC and one Level $n-1$ quadcode (QC) orderings. Find a recursive expression for the ADFSD based on the observation that the total number of neighbor pairs equals the sum of the number of neighbor pairs in the three Level $n-1$ orderings, plus the additional neighbor pairs where these three meet. The sum of the rank differences at Level n will be equal to the sum of rank differences of each of the two Level $n-1$ SQC orderings, plus that of the Level $n-1$ QC ordering, plus the rank differences across the edges where the three Level $n-1$ orderings meet. Given this recursive expression, the proposition follows readily. For large n, this result is close to $(34/63)2^n$.

Proof of Proposition 13

In all of the curves, each Level n subdivision is made up of four Level $n-1$ subdivisions. The total number of pairs of cell neighbors at Level n is therefore four times the number at Level $n-1$, plus extra pairs at the three boundaries where the four subdivisions link. We observe that each of these boundaries contains edges from 2^{n-1} subcells. Therefore, $p(n) = 4 \cdot p(n-1) + 3 \cdot 2^{n-1}$. From this recursive equation, and the fact that $p(1) = 3$, for both curves, the result follows by induction.

Proof of Proposition 14

Level n of the Sierpinski curve can be divided into four Level $n-1$ curves, or *quadrants*. Let us call them $Q0, Q1, Q2, Q3$. The four quadrants meet at three boundaries. The following pairs of quadrants have boundaries: $(Q0, Q1)$, $(Q1, Q2)$, and $(Q2, Q3)$. In addition, it is easy to verify that, for each pair, edges from 2^{n-1} subcells within each quadrant lie on the boundary.

Let $S(n)$ stand for the total sum of the differences across all shared edges at Level n. Clearly, $S(n)$ equals $4 \cdot S(n-1)$, plus the sum of all the differences across the three boundaries between adjacent quadrants. We now look at each of these boundaries.

$Q1$ and $Q2$ are adjacent at their respective hypotenuses. Therefore, the same relative sets of cells from each quadrant are adjacent. The only difference is that the cells from $Q2$ are offset by 4^{n-1}, the number of cells in a quadrant. (This is true by the self-similarity of these structures.) Therefore, the sum of the differences across this boundary is $2^{n-1} \cdot 4^{n-1}$ (the number of adjacent cells, times the offset of the two quadrants).

The remaining two boundaries, $(Q0, Q1)$ and $(Q2, Q3)$, are equivalent cases. We just look at the first case. $Q0$ and $Q1$ are unfortunately not adjacent at their respective hypotenuses. But if we further divide each into octants, we find that we have two octants adjacent at their respective hypotenuses (one octant from $Q0$ and one from $Q1$). In this case, the same relative set of cells from each octant is adjacent, and the cells are offset by $4^n/8$, the number of cells in an octant. Therefore, the sum of the differences across this boundary is $2^{n-1} \cdot 4^n/8$ (the number of adjacent cells, times the offset of the two octants). Now we have

$$\begin{aligned} S(n) &= 4S(n-1) + 2^{n-1}4^{n-1} + 2 \cdot 2^{n-1}\left(\frac{4^n}{8}\right) \\ &= 4S(n-1) + 2^n 4^{n-1}. \end{aligned}$$

From this equation, and the observation that $S(1) = 3$, we can find the expression

$$S(n) = 4^{n-1}(2^{n+1} - 1),$$

by standard induction arguments.

Dividing by the number of pairs of neighboring cells (Proposition 13), we find

$$ADFSD_{\text{ssfc}} = \frac{2^n(2^n - 2^{-1})}{3(2^n - 1)}.$$

This is close to $(1/3)2^n$, for large n.

Proof of Proposition 15

Level n of the Quaternary curve can be divided into four Level $n-1$ curves, or *quadrants*. Let us call them $Q0, Q1, Q2, Q3$. The four quadrant meets at three boundaries. The following pairs of quadrants have boundaries: $(Q0, Q1)$, $(Q1, Q2)$, and $(Q1, Q3)$. It is easy to verify that, for each pair, edges from 2^{n-1} subcells within each quadrant lie on the boundary.

Let $S(n)$ stand for the total sum of the differences across all shared edges at Level n. Clearly, $S(n)$ equals $4 \cdot S(n-1)$, plus the sum of all the differences across the three boundaries between adjacent quadrants. We now look at each of these quadrant boundaries.

Each of quadrants $Q1$ and $Q3$ has the same set of cells (relatively speaking) having edges on their boundary. The only difference is that the cells from $Q2$ are offset by $2 \cdot 4^{n-1}$, the number of cells in a quadrant. (In other words, by subtracting $2 \cdot 4^{n-1}$ from the labels of each cell in $Q3$ that lie on the adjacent boundary, one would arrive at the set of labels of $Q1$ cells on the boundary.) Therefore, the sum of the differences across this boundary is $2^{n-1}(2 \cdot 4^{n-1})$ (the number of adjacent cells, times the offset of the two quadrants).

The remaining two boundaries, $(Q0, Q1)$ and $(Q1, Q2)$, are equivalent cases. We just look at the first case. $Q0$ and $Q1$ are unfortunately not adjacent at the same relative sides. But if we further divide each quadrant into four subquadrants (that is, sixteenths), we have two subquadrants from $Q0$ adjacent to the two subquadrants from $Q1$. Again, the same relative set of cells from each is adjacent. The offsets are a little more complicated than in the case of the Sierpinski curve, where the two octants were adjacent in the ordering. Now we have the $3rd$ sixteenth (from $Q0$) adjacent to the $4th$ sixteenth (from $Q1$), and the $2nd$ sixteenth (from $Q0$) adjacent to the $7th$ sixteenth (from $Q1$). In this case, each sixteenth is offset by $4^n/16 = 4^{n-2}$, the number of cells in $1/16^{th}$ of the subdivision. Further, the number of cells along each of the boundaries between sixteenths is 2^{n-2}. Therefore, the sum of the differences across this boundary is $2^{n-2}(5 \cdot 4^{n-2}) + 2^{n-2}(1 \cdot 4^{n-2})$.

Summing up, we have

$$\begin{aligned} S(n) &= 4S(n-1) + 2 \cdot 2^{n-1}4^{n-1} + 2(6 \cdot 2^{n-2}4^{n-2}) \\ &= 4S(n-1) + 7 \cdot 2^{n-2}4^{n-1}. \end{aligned}$$

From this recursive equation, and the observation that $S(1) = 4$, we can find the sum of rank differences between neighboring cells,

$$S(n) = 4^{n-1}(7 \cdot 2^{n-1} - 3),$$

by standard induction arguments.

Dividing by the number of pairs of neighboring cells (Proposition 13), we find

$$ADFSD_{\text{qsfc}} = \frac{7 \cdot 2^n(2^n - 6)}{12(2^n - 1)}.$$

This is close to $(7/12)2^n$, for large n.

Proof of Proposition 18

For Dutton-type orderings at Level $n \geq 2$, there are $4^n - 1$ sequential cell pairs. It is straightforward to show that 4^{n-1} such pairs are edge-adjacent, and $9 \cdot 4^{n-2}$ are vertex-adjacent, and the proposition follows directly.

Proof of Proposition 19

For Otoo and Zhu (SQC) orderings at Level $n \geq 2$, there are $4^n - 1$ sequential cell pairs. It is straightforward to show that $3 \cdot 4^{n-1} - 2^{n-1}$ such pairs are edge-adjacent, and $3 \cdot 4^{n-2} + 2^{n-2}$ are vertex-adjacent, and the proposition follows directly.

Acknowledgments

We appreciate the support of The Logistics Institute at the School of Industrial and Systems Engineering at Georgia Tech; the Office of Naval Research through Grant #N00014-89-J-1571; and the National Science Foundation through Grant #DMI-9908313.

References

Abel, D. J. and Mark, D. M., 1990, A comparative analysis of some two-dimensional orderings, *International Journal of Geographical Information Systems* **4**(1), 21–31.

Barrera, R., 1989, A shortest path method for hierarchical terrain models, *Proc. Auto Carto 9, April 2–7, 1989*, Baltimore, Maryland, pp. 156–163.

Bartholdi, III, J. J. and Goldsman, P., n.d.a, A continuous spatial index of a triangulated surface. Submitted.

Bartholdi, III, J. J. and Goldsman, P., n.d.b, Vertex-labeling algorithms for the hilbert spacefilling curve, *Software—Practice and Experience* . To appear.

Bartholdi, III, J. J. and Platzman, L. K., 1982, An $O(n \log n)$ planar travelling salesman heuristic based on spacefilling curves, *Operations Research Letters* **1**, 121–125.

Bartholdi, III, J. J. and Platzman, L. K., 1988, Heuristics based on spacefilling curves for combinatorial problems in Euclidean space, *Management Science* **34**(3), 291–305.

Bartholdi, III, J. J., Platzman, L. K., Collins, R. L. and Warden, W. H., 1983, A minimal technology routing system for meals-on-wheels, *Interfaces* **13**, 1–8.

Croft, H. T., Falconer, K. J. and Guy, R. K., 1991, *Unsolved Problems in Geometry*, Springer-Verlag, New York.

Davies, H. L., 1967, Packings of spherical triangles and tetrahedra, *in* W. Fenchel (ed.), *Proc. of the Colloquium on Convexity, 1965*, Kobenhavns Univ. Mat. Inst., pp. 42–51.

Dutton, G., 1984, Geodesic modelling of planetary relief, *Cartographica* **21**(2&3), 188–207.

Dutton, G., 1989, Planetary modelling via hierarchical tessellation, *Proc. Auto Carto 9, April 2–7 1989*, Baltimore, Maryland, pp. 462–471.

Dutton, G., 1990, Locational properties of quaternary triangular meshes, *Proc. of the 4th International Symposium on Spatial Data Handling, July 23–27 1990*, Vol. 2, Zurich, Switzerland, pp. 901–910.

Dutton, G., 1996, Improving locational specificity of map data—a multi-resolution, metadata-driven approach and notation, *International Journal of Geographical Information Systems* **10**(3), 253–268.

Fekete, G., 1990, Rendering and managing spherical data with sphere quadtrees, *Proc., Visualization '90, Oct. 23–26 1990*, San Francisco, CA, pp. 176–186.

Fekete, G. and Davis, L. S., 1984, Property spheres: A new representation for 3-D object recognition, *Proc. Workshop on Computer Vision Representation and Control, April 30–May 2 1984*, Annapolis, MD, pp. 192–201.

Goodchild, M. F., 1989a, Optimal tiling for large cartographic databases, *Proc. Auto Carto 9, April 2–7 1989*, Baltimore, Maryland, pp. 444–451.

Goodchild, M. F., 1989b, Tiling large geographical databases, *in* G. Goos and J. Hartmanis (eds), *Proc. SSD '89, July 17–18 1989*, Vol. 409 of *Lecture Notes in Computer Science*, Springer-Verlag, Santa Barbara, CA, pp. 137–146.

Goodchild, M. F. and Grandfield, A. W., 1983, Optimizing raster storage: An examination of four alternatives, *Proc. Auto Carto 6*, Vol. 2, Ottawa, pp. 400–407.

Goodchild, M. F. and Shiren, Y., 1989, A hierarchical spatial data structure for global geographic information systems, *Technical report*, National Center for Geographic Information and Analysis. Technical Paper 89-5.

Goodchild, M. F. and Shiren, Y., 1990, A hierarchical data structure for global geographic information systems, *Proc. of the 4th International Symposium on Spatial Data Handling, July 23–27 1990*, Vol. 2, Zurich, Switzerland, pp. 911–917.

Goodchild, M. F. and Shiren, Y., 1992, A hierarchical spatial data structure for global geographic information systems, *CVGIP: Graphical Models and Image Processing* **54**(1), 31–44.

Hilbert, D., 1891, Ueber die stetige abbildung einer linie auf ein flachenstueck, *Math. Ann.* **38**.

Laurini, R. and Thompson, D., 1992, *Fundamentals of Spatial Information Systems*, Academic Press, Ltd., San Diego, CA.

Mark, D. M. and Lauzon, J. P., 1984, Linear quadtrees for geographic information systems, *Proc. International Symposium on Spatial Data Handling, Aug 20–24, 1984*, Zurich, Switzerland, pp. 412–430.

Mark, D. M. and Lauzon, J. P., 1985, Approaches for quadtree-based geographic information systems at continental or global scales, *Proc. Auto-Carto 7, March 11–14, 1985*, Washington, D.C., pp. 355–364.

Nulty, W. G., 1993, *Geometric Searching with Spacefilling Curves*, PhD thesis, Georgia Institute of Technology, Atlanta, GA.

Otoo, E. J. and Zhu, H., 1993, Indexing on spherical surfaces using semi-quadcodes, *in* D. Abel and B. C. Ooi (eds), *Advances in Spatial Databases—Third Annual Symposium, SSD '93*, Springer-Verlag, Singapore, pp. 510–529.

Platzman, L. K. and Bartholdi III, J. J., 1989, Spacefilling curves and the planar travelling salesman problem, *JACM* **36**(4), 717–737.

Sagan, H., 1994, *Space-Filling Curves*, Springer-Verlag, New York.

Samet, H., 1989, Hierarchical spatial data structures, *in* G. Goos and J. Hartmanis (eds), *Proc. SSD '89, July 17–18, 1989*, Vol. 409 of *Lecture Notes in Computer Science*, Springer-Verlag, Santa Barbara, CA, pp. 193–212.

Samet, H., 1990, *The Design and Analysis of Spatial Data Structures*, Addison-Wesley, Reading, MA.

Samet, H., 1995, Spatial databases, tutorial, *SSD'95, Aug. 6. 1995*, Portland, Maine, USA.

Scarlatos, L. and Pavlidis, T., 1992, Hierarchical triangulation using cartographic coherence, *CVGIP: Graphical Models and Image Processing* **54**(2), 147–161.

Sierpinski, W., 1912, Sur une nouvelle courbe qui remplit toute une aire plaine, *Bull. Acad. Sci. Cracovie, Serie A* pp. 462–478.

Sotomayor, D. L. G., 1978, Tessellation of triangles of variable precision as an economical representation for DTM's, *Proc. Digital Terrain Models Symposium, May 9–11, 1978*, St. Louis, Missouri, pp. 506–515.

Weibel, R. and Dutton, G., 1999, Generalising spatial data, *in* P. A. Longley, M. F. Goodchild, D. J. Maguire and D. W. Rhind (eds), *Geographical Information Systems*, second edn, John Wiley & Sons, New York, chapter 10.

Wenninger, M. J., 1979, *Spherical Models*, Cambridge University Press.

White, D., Kimerling, A. J., Sahr, K. and Song, L., 1998, Comparing area and shape distortion on polyhedral-based recursive partitions of the sphere, *International Journal of Geographical Information Science* **12**(8), 805–827.