INDEXING OF THE DISCRETE GLOBAL GRID USING LINEAR QUADTREE

Jianjun Bai[a], Xuesheng Zhao [a,b], Jun Chen[b]

[a].China University of Mining and Technology (Beijing), Beijing 100083,China;

[b]. National Geomatics Center of China, Beijing 100044, China

KEY WORDS： Indexing, The discrete global grid, Linear quadtree, Neighbour- finding, Recursive subdivision, Diamond

ABSTRACT:

In recent years a method of recursive subdivisions of the triangular faces of the octahedron or icosahedron has been developed for approximating to the surface of the earth. But the triangle-based discrete grids produced by this method are complicated in geometry structure, and is difficult to make such geographical operations as neighbor-finding, spatial searches and so on. In this paper we conceives of the surface of the octahedron as composed of pairs of adjacent triangles, or diamond, that tessellate the surface, and thus creates nested diamond subdivision of the ellipsoidal surface by quadtree recursive partition. The quadtree Morton coding system is used as the index for addressing the diamonds and for linearizing storage that preserve a high degree of spatial locality. And a method of finding neighbor, ancestors and descendants also is developed. Based on this we further develop an index for addressing the triangle and a neighbor-finding method. The addressing system exhibits a high degree of regularity that makes it possible to develop very efficient algorithms for common spatial database and geometric operations

1. INTRODUCTION

In order to meet the needs of applications in dynamic modeling, data organizing, information storage and display, global grid model that uses the subdivided surface of Platonic solids, or regular polyhedron, as the approximations to the surface of the earth has been studied extensively. Among them quaternary triangular meshes (QTM) developed by the partition of either the octahedron or icosahedron is one of the representative methods. Because it uses triangle as the basic cell to organize spherical data that fall into this area, many spatial analysis methods must be completed based on the triangle. However the geometric structure of the triangle meshes is very complex, it has asymmetry and uncertainty orientation that make it difficult to implement such spatial operation as adjacent analysis, spatial query, data update and so on.

In this paper we conceive of the surfaces of the octahedron and icosahedron as composed of pairs of adjacent triangles, or diamonds, that tessellate, or cover the surface of the earth. The surface of the earth can be approximated through the diamond subdivision. The spherical data is organized as diamond cell, thus the linear quadtree Morton addressing system can be used to label and index the cell. Based on this we further develop an index for addressing the triangle and a neighbor-finding method of the triangle. The addressing system exhibits a high degree of regularity that makes it possible to develop very efficient algorithms for common spatial database and geometric operations.

2. PREVIOUS WORK

Dutton[1990] and Fekete[1990] introduce a “floating” labeling scheme in which the labeling of the child triangle at each level varies based on the orientation of the parent triangle. Indexing and neighbor-finding of the triangle also be introduced based on this labeling technique. The advantage of this labeling technique is that the path components of all location codes that correspond to the neighbors of a particular triangle differ by one directional code, at the expense of added complexity for the neighbor-finding process. Moreover, this technique has the further disadvantage that finding neighbors that lie on different faces of the polyhedron is much harder.

Goodchild and Yang[1992, Lee and Samet[2000] and ZHAO Xuesheng [2002] adopt a similar labeling scheme in which the labeling of the child triangle fixes according to the location of it in the parent triangle (left, right, middle and top). The neighbor-finding methods used in Goodchild and Yang[1992] and ZHAO Xuesheng [2002] have a worst-case execution time proportional to the maximum level of decomposition. Although based on the same principle as this methods, the algorithms of the Lee and Samet[2000] use binary arithmetic, and have a worse-case constant time complexity.

Otoo and Zhu [1993] use a SQC (Semi-Quadcodes) labeling scheme in which two triangles that make up each square can be distinguishes through add an additional bit at the end of the Quadcodes. This makes it possible to use algorithms and techniques developed for Quadtree Square meshes, including a variant of the worst-case constant time neighbor-finding algorithms, but his algorithms can not deal with the case when the neighbor triangle across the different face of the polyhedron.

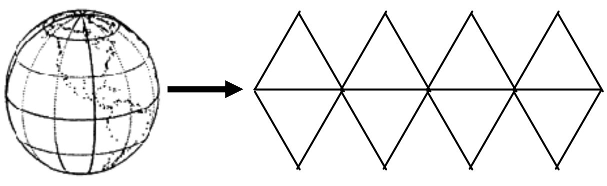

Figure 1. Four base diamond of the octahedron

3. GLOBAL GRIDS BASED ON THE QUATERNARY TRIANGULAR AND DIAMOND SUBDIVISIONS

Similar to White [8], we conceive of the surfaces of the octahedron and icosahedron as composed of pairs of the adjacent (south and north) triangles, or diamonds, that tessellate, or cover the surface, so the octahedron has four base diamonds (we also refer to it as quadrant). The octahedron that made up by four diamonds is a coarsest approximation to the surface of the earth.

Each triangle in the octahedron can be divided into four smaller equilateral triangles by breaking each edge into 2 pieces and connecting the break points with lines. which is referred to as QTM(quaternary triangular mesh)subdivision. Recursively subdividing the triangles thus obtained in the same manner yields QTM (figure 2); like the quadtree subdivision of the square, each quadrant (base diamond) in the octahedron can be divided into four smaller diamonds (figure 2). These two kinds of subdivision are essentially the same, so the QTM can be regarded as diamond meshes. The surface of the earth can be represented as a quadtree that the root (corresponding to the surface of the earth) has four children node (four base diamonds), and the internal node has four children node.

It must be illustrated that the diamonds on the polyhedron are bent, and the four points that form the diamond are not in one plane, but for data structure they can be considered entire. Similar to the square meshes, the geometric structure of diamonds meshes is much more simpler than that of triangular mesh. Unlike the triangular mesh (has up and down triangle), it has uniform orientation, thus make the spatial operation especially neighbor finding easier.

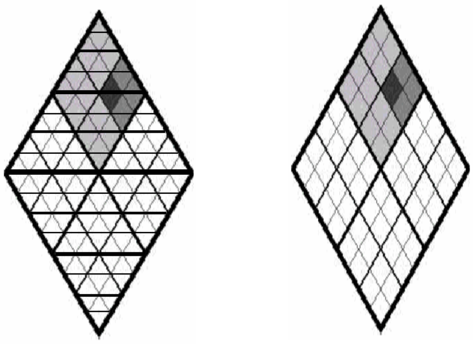

Figure 2. QTM and the diamond tessellation at the third level

4. INDEXING AND CODING OF DIAMOND BASED ON LINEAR QUADTREE

The surface of the earth can be represented as a linear quadtree that there need only the location of leaves to be registered in the storage process. The leaves can be labeled according to the Z spacefilling curve (figure 4). Each diamond is assigned a quadcode D and Morton codes M. The code of a diamond L can be represented as DM, where D is the quadcode of the quadrant, and M is the Morton codes of the diamond in the same quadrant. The quadrant is assigned a numerical label 0,1,2 or 3 (The quadcode D) according its location on the surface of the earth. It is illustrated as follows:

D= 0, $90° > \lambda \geq 0°$

D= 1, $180° > \lambda \geq 90°$

D= 2, $270° > \lambda \geq 180°$

D= 3, $360° > \lambda \geq 270°$

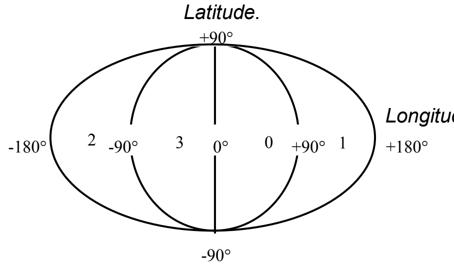

Figure 3. The coding of four base diamonds

Morton codes are used to label the different diamond produced by the subdivision within the same quadrant. It has two kinds of form: quaternary Morton code and decimal Morton code.

Each diamond will be substituted by four smaller diamonds when performs subdivision. These four newly produced diamonds can be labeled respectively through adding a additional digit 0, 1, 2 and 3 according to their location (left, down, up and right) in the parent diamond (figure 4). Thus, the Morton code of a diamond consists of a sequence of numbers (0,1,2or3), where the length of the sequence represents the partition level of the base diamonds (quadrant). Each digital of the Morton code is the Arabic digital no larger than 3.

$$M_Q = q_1 q_2 q_3 \cdots q_k = q_1 \cdot 10^k + q_2 \cdot 10^{k-1} + \cdots + q_k \quad (1)$$

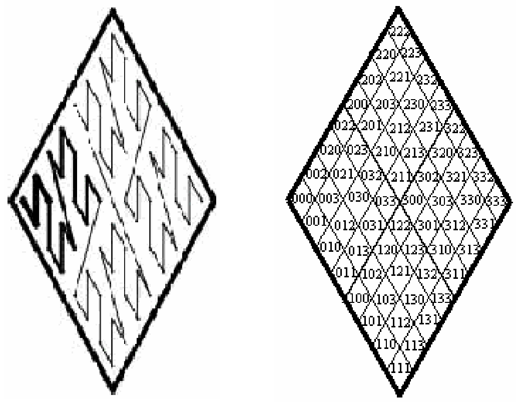

Figure 4. The Morton code and Z space filling curve of diamonds

5. NEIGHBOR FINDING OF DIAMOND

It is easy to determining the Morton codes of either the sons or the parent of a diamonds according to the properties of the Morton codes. The sons of a diamond are determined by appending the digits 0,1,2 and 3 at the end of the Morton codes, and the address of the parent of a diamond is obtained by simply discarding the trailing quaternary digit.

In order to determine adjacent neighbors of the diamond, firstly we must take the conversions between the Morton codes and row/column number of the diamond. The conversion algorithm between the Morton codes and row/column number of the diamond are introduced in detail in paper [11]. Figure 5 show the correlation between them.

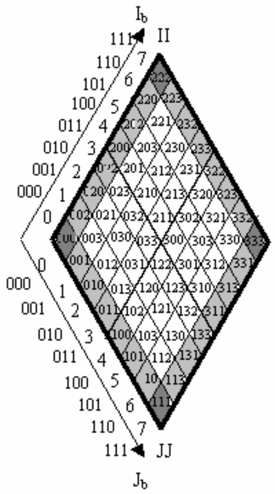

Figure 5. The correlation between quaternary Morton code and row/column number

The neighbor finding of the diamond is much more easier than that of the triangle, this is determined by the properties of the diamond mesh. Because the triangle cell in the triangle mesh has not uniform orientation and the adjacent neighbor of it varies according to its location in the mesh. Neighbor finding is different for either different orientated triangle or different located triangle in the mesh. Diamond mesh has several properties such as radial symmetry, translation congruence and uniform orientation, unlike triangles, thus makes some spatial operations such as neighbor-finding much more easily.

There are three cases in the neighbor-finding of the diamond: the diamond located in the inside of the base diamond (quadrant), the diamond located in the boundary of the base diamond and the diamond located in the corner of the base diamond.

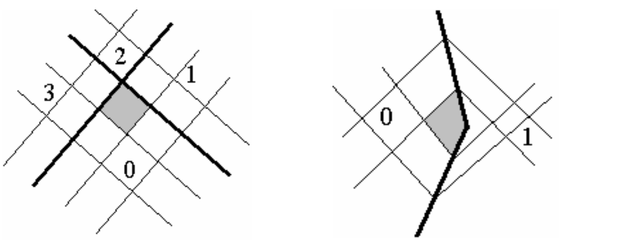

Figure 6. Adjacent neighbors of the diamond located on the corner of the base diamond

Algorithm for determining the edge adjacent neighbors is as follows (the algorithm can be extended to computed the address of the vertex adjacent neighbors as well):

```
EdgeDiamondAdjacent (QDcode DM, Direction Dir)
{
  D←PrefixD(DM); // extract the started digit
  M←DelePrefixD(DM); // delete the start digit
  (i,j) ←F-1(M)   // conversion from Morton code to row/column numner
  switch(Dir)
  {
    EN:      // the northeast neighbor
      if(i<I)   i++;   // located in same base diamond
      else             // located in different base diamond
      {if(d=3) d=0; else d++
      k=j;j=0;i=I-k;} break;
    ES:      // the southeast neighbor
      f(i>0) i--;   // located in same base diamond
      else             // located in different base diamond
      {if(d=0) d=3; else d--
      k=j;j=I;i=I-k;} break;
    WN:               // the northwest neighbor
      if(j>0) j--;   // located in same base diamond
      else             // located in different base diamond
      {if(d=0) d=3; else d--
      k=i;i=I;j=I-k;} break;
    WS:        // the southwest neighbor
      if(j<I) j++; // located in same base diamond
      else            // located in different base diamond
      {if(d=3) d=0; else d++
      k=i;i=0;j=I-k;} break;
  }
  M←F(i,j); // conversion from row/column number to Morton code
  DM←AppendToD(M,D); // append the quadcode of the base diamond
  Return(DM);
  }
```

6. NEIGHBOR FINDING OF TRIANGLE

Diamond tessellations are compatible with tessellations of triangles in nature. The diamond can be regarded as the merging of two triangles. These two triangles made up the diamond can be distinguished through appending a digit 0 or 1 to the end of code of the diamond. As showed in figure 7, the code of triangle ended with"0" indicates that it has a south edge-adjacent neighbor, and the code of triangle ended with "1" indicates that it has a north edge-adjacent neighbor within the same diamond. The suffixes of the code of triangle indicate the triangle is either up or down triangle (figure 7). The code of triangle traces a zigzag course through the triangles at any given level (figure 7). Based on the algorithm of neighbor finding of the diamond, we develop an algorithm for determining the edge adjacent neighbors of triangle. The same strategy can be used to compute the address of the vertex adjacent neighbors as well.

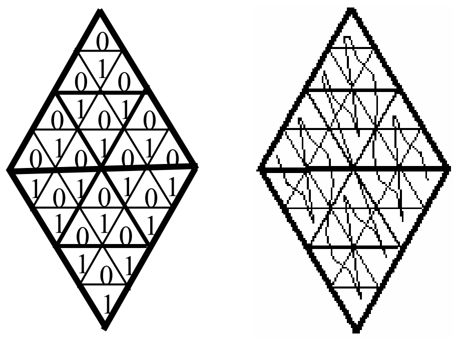

Figure 7. Labeling and indexing of two triangles made up the diamond

Algorithm for determining the edge adjacent neighbors of the triangles is as follows:

EdgeTriangleAdjacent (QTcode TM, Direction Dir)

```
{
DM←DeleSuffix(TM);    // delete the terminal digit
M←DelePrefixD(DM); // delete the started digit
(i,j) ←F-1(M)          // conversion from Morton code of the diamond to row/column number
T←SuffixD(TM); // extract the orientation of the triangle
   switch(Dir)
   {
   EAST:              // the east neighbor of the triangle
   if(T=0)
      DM←EdgeDiamondAdjacent(DM,EN);
      if(i<I)   TM←AppendTODM(DM,1);
         else
            TM←AppendTODM(DM,0);
         else
            DM←EdgeDiamondAdjacent(DM,ES);
         if(j<I)   TM←AppendTODM(DM,0); else
          TM←AppendTODM(DM,1);
         break;
                  WEST:        // the west neighbor of the triangle
                        if(T=0)
                        DM←EdgeDiamondAdjacent(DM,WN);
                        if(j>0)   TM←AppendTODM(DM,1);
                        else
                              TM←AppendTODM(DM,0);
                        else
                           DM←EdgeDiamondAdjacent(DM,WS);
                        if(i>0)   TM←AppendTODM(DM,0);
                        else
                              TM←AppendTODM(DM,1);
                        break;
                  INVERT:if(T=0)
                              TM←AppendTODM(DM,1);
                        else
                              TM←AppendTODM(DM,0);
                        break;}
            return(TM)
            }
```

7. CONCLUSIONS

Data organization based on the diamond tessellation has several advantages. Diamond geometry is simpler than triangles. Like the regular grid, the diamond cell has uniform orientation, radial symmetry and translation congruence, thus make it much more easier to complete nearly all the spatial operations of the discrete data a spherical surface. The diamond hierarchy is nested that make it convenient for data organization and compressed storage. We use a labeling scheme in which two triangles that make up each diamond can be distinguishes through add an additional digit to the end of the Quadcodes of the diamond. This makes it possible to extend algorithms and techniques developed for Quadtree Square meshes to adapt to the triangular meshes.

REFERENCES

Bartholdi. J. III and Goldsman P., 2001. Continuous Indexing of Hierarchical Subdivisions of the Globe. *Int. J. Geographical Information Science*, 15(6), pp.489-522.

Dutton G.,1990. Locational Properties of Quaternary Triangular Meshes. In: *the Fourth International Symposium on Spatial Data Handing* (Zurich, Switzerland July 1990). pp. 901-910.

Fekete,G., 1990. Rendering and managing spherical Data With Sphere Quadtrees. In: *Conference on Visualization'90* (san Francisco, CA,Oct.1990).

Goodchild M.F. and Yang S., 1992. A Hierarchical Data Structure for Global Geographic Information Systems. *Computer Vision and Geographic Image Processing*, 54(1), pp.31-44.

Kevin Sahr,Denis White,and A.Jon Kimerling.,2003. Geodesic Discrete Global Grid Systems. *Cartography and Geographic Information Science*,30(2), pp. 121-134.

Lee M. and Samet H., 2000, Navigating through Triangle Meshes Implemented as Linear Quadtrees. *ACM Transactions on Graphics*, 19(2), pp. 79–121.

Otoo E. and Zhu H.,1993. Indexing on Spherical Surfaces Using Semi-Quadcodes, In: Abel J. and Beng C. O.(Eds.), *Advances in spatial Databases 3th International Symposium, SSD`93*, Singapore, pp. 509-529.

White D. and Kimmerling A.J., 1998. Comparing Area and Shape Distortion on Polyhedral Based Recursive Tessellations of the Sphere. *International Journal of Geographical Information Science*. 12(8), pp. 808-827.

White. D., 2000. Global Grids From Recursive Diamond Subdivisions of The Surface of an Octahedron or Icosahedron. *Environmental Monitoring and Assessment*, 4(1), pp. 93-103.

Zhao xuesheng, 2002. Spherical Voronoi Data Model Based on QTM. PH.D dissertation, China University of Mining and Technology (Beijing), Beijing, pp.105 in chinese.

ACKNOWLEDGEMENTS

This work described in the paper is supported by the National Natural Science Foundation of China (Under grant No.40471108)